Development of Smoothed Particle Hydrodynamics for Flow in Complex Geometries and Application of Open Source Software for the Simulation of Turbulent Flow

Obeidat, Anas Hassan MohD

Publication date:
2014

Document Version
Publisher's PDF, also known as Version of record

Development of Smoothed Particle Hydrodynamics for Flow in Complex Geometries and Application of Open Source Software for the Simulation of Turbulent Flow

Anas Obeidat

A dissertation submitted for the degree of Doctor of Philosophy

Section of Fluid Mechanics
Department of Mechanical Engineering
Technical University of Denmark
Kongens Lyngby
June 2014

Acknowledgements

This dissertation is submitted to fulfil the last requirements for obtaining a Ph.D. degree in Mechanical engineering at the Technical University of Denmark (DTU). The Ph.D. Project carried out at section of Fluid Mechanics at DTU under the supervision of Professor Ph.D. Jens Honoré Walther.

First of all i would like to give my biggest thanks and appreciation to my tough supervisor Jens Honoré Walther 'The BOSS', for his support, the knowledge he shared with, the friendship, and for keeping me on track and not being easy with me.

I want to thank all of my colleagues at DTU, section of fluid mechanics and wind energy, building 403, for being more than colleagues, but friends, for the amazing atmosphere we had while working there, and the tough academic discussions. Special thanks to Sajjad Haidar, for his support at the beginning of my carrier at DTU. To all my close friends that i did meet in Copenhagen, I would like to show my appreciation and thanks to all of you for the awesome time we had together.

Furthermore I would like to show my gratitude to Ivo Sbalzarini and the MOSAIC group at ETH-Zurich for welcoming me during my stay there, and for our collaboration.

I would like to give very special thanks to two of my closest friends, Ömer for being more than a friend, for all the help, good talks and academic discussions we shared in the past seven years. And to David for all the good times we did share.

Abstract

Turbulence modelling is a key issue in many industrial application, as the computational power of direct numerical simulation (DNS) is insufficient to deal with complex flow structures with high Reynolds number. Also in Industrial applications often involve turbulent flow in complex geometries. Thus developing a computational method which can deal with complex fluid structure, simulate complex geometers that change topology is particular challenging as the connectivity of the computational domain may change dynamically, and still efficient is important.

In this thesis we are presenting a remeshed particle-mesh method, the method involves three-dimensional compressible turbulent flow modelling, and coupled with an immersed boundary technique to deal with the complex solid obstacles.

This dissertation is composed of three parts.

In combustion engines the scavenging process in two-stroke marine diesel engines removes combustion gases from the engine cylinder and fills up the cylinder with the fresh air charge for the next cycle. Understanding the scavenging flow is crucial for the development of such engines, since it affects fuel consumption, engine cooling and production of pollutants.

We consider a state-of-the art eulerian methods to study the turbulent flow in a model diesel engine. the goals of this study include validation of large eddy simulations (LES) turbulence models.

We study the effect of piston position on the in-cylinder swirling flow in a simplified model of a large two-stroke marine diesel engine. To be able to simulate

the complex flow structures with high Reynolds number, large eddy simulations (LES) with four different models for the turbulent flow are used: a one-equation model, a dynamic one-equation model, a localised dynamic one-equation model and a mixed-scale model. The added LES numerical model makes it possible to the system to approximate the unresolved small flow structures.

Simulations are carried out for two different geometries corresponding to 100 % and 50 % open scavenge-ports. It is found that the mean tangential profile inside the cylinder changes qualitatively with port closure from a Lamb-Oseen vortex profile to a solid body rotation while the axial velocity changes from a wake-like profile to a jet-like profile. The numerical results are compared with particle image velocimetry measurements and in general we find a good agreement. Considering the complexity of the real engine, we designed the engine model using the simplest configuration possible. The setup contains no moving parts, the combustion is neglected and the exhaust valve is discarded. Studying the flow in a simplified engine model, the setup allows studies of fundamental aspects of swirling flow in a uniform scavenged engine. Comparing the four turbulence models, the local dynamic one-equation model is found to give the best agreement with the experimental results.

The second part of the thesisWe present a remeshed particle-mesh method for the simulation of three-dimensional compressible turbulent flow. The method is related to the mesh free smoothed particle hydrodynamic (SPH) method, but the present method introduces a mesh for efficient calculation of the pressure gradient, and laminar and turbulent diffusion. In addition, the mesh is used to remesh (reorganise uniformly) the particles to ensure a regular particle distribution and convergence of the method. The accuracy of the presented methodology is tested for a number of benchmark problems involving two- and three-dimensional Taylor-Green flow, thin double shear layer, and three-dimensional isotropic turbulence. Two models were implemented, direct numerical simulations, and Smagorinsky model. Taking advantage of the Lagrangian advection, and the finite difference efficiency, the method is capable of providing quality simulations while maintaining its robustness and versatility.

Finally we couple the remeshed particle-mesh method with Brinkman penalisation resulting an implicit boundary particle method with background mesh adaptation. We use a Brinkman penalisation to represent the boundary of the domain and a remeshed particle method to simulate viscous flow with high Reynolds numbers. A penalty term is added to the Navier-Stokes equations to impose the boundary conditions. The boundary conditions are enforced to a specific precision with no need to modify the numerical method or change the grid, achieving an implicit approach for flow around complex boundaries/geometries simulation. The main idea of the Brinkman penalisation method is to model the solid obstacle as a porous medium. The governing equations for the compressible

fluid and penalised Navier-Stokes for the porous medium are solved simultaneously, without need for interface conditions. The accuracy of the method is tested for a number of benchmark problems starting with simple cases like, a periodic laminar flow inside a pipe (Poiseuille flow), to more complex problems such as the lid-driven cavity with high Reynolds number, the reflection and propagation of shock compressible wave. The remeshed particle-mesh method with Brinkman penalisation provides a good quality simulation and the results show good agreement with analytical or reference solutions.

Contents

Chapter 1

Introduction

We are surrounded by fluids, we encounter fluid flows in everyday life, meteorological phenomena (rain, wind, hurricanes), environmental hazards (air pollution), interaction of various objects with the surrounding fluids (combustion, heating, ventilation), processes in human body (blood flow, breathing), and chemical reactors etc...

The fluid flow varies from laminar to highly turbulence flow. Engineers and researchers invest time and money to understand the physical structure of the flow; to create accurate observations and measurements that will later help to improve their designs or modify their equations.

Before the leap in computational science, the manufacturing of big computers, and parallel computing, the study of fluid dynamics was limited to theoretical and experimental techniques.

Experiments describe the flow phenomena using measurements, however it suffers from several limitations that limit the ability to understand the structure of the flow. Experimental techniques are limited to laboratory scale models, limited number of measurement points, and small number of problems.

Computational fluid dynamic (CFD) is playing an important role in understanding the complex structure of the fluid, giving a qualitative and quantitative

prediction of fluid flows where the experiments fail to.

CFD is highly efficient in term of accuracy and simulation time when it is used to simulate simple flow, which makes it a competitive tool with experiments. However in the case of complex flow which has a large amount of information and small structures as in turbulent flow, CFD is limited in terms of accuracy, time, and computationally impossible. It is essential to add numerical modelling to the system to approximate the unresolved phenomena, like turbulence modelling, complex meshes, two phase flow, and other models. The included model has one main purpose, to predict quantities of interest, that will enhance the engineering designs of the system.

To address these challenges the present thesis present results from simulations of three dimensional turbulence flow using an open source code OpenFOAM, and an in-house code using as a part of the The Parallel Particle Mesh (PPM) [98], where the Navier-Stokes equations are discretised and different turbulence models are used.

OpenFOAM is used to study the effect of piston position on the in-cylinder swirling flow in a simplified model of a large two-stroke marine diesel engine. The effect of the piston position is being examined, and Large Eddy Simulations with four different models for the turbulent flow were implemented using the OpenFOAM platform.

In the second part of the thesis we present a remeshed particle-mesh method for the simulation of three-dimensional compressible turbulent flow, the method is related to the mesh free smoothed particle hydrodynamic (SPH), but for efficient calculation of the turbulent diffusion, and the pressure gradient a mesh is introduced, also the mesh is used to ensure the convergence of the method, and to maintain a regular distribution of the particles by reinitialising them on a uniform mesh, the method is called hybrid remeshed smooth particle method (hrSPH).

In the third part we present Brinkman penalisation method for imposing complex boundaries in the flow simulation, which is essential in several engineering applications. The Brinkman method belong to the immersed boundary methods family. We coupled our hrSPH for compressible flow with Brinkman method to be able to simulate flow cases where complex geometries are needed.

The thesis is structured as follows:

Part I: Large eddy simulations of the influence of piston position on the swirling flow in a model two-stroke diesel engine

In this part we focus on an industrial approach of computational science and turbulence modelling application.

We study the effect of piston position on the scavenging and swirling flow in a simplified model of a large two-stroke marine diesel engine cylinder.

We consider two cases of static piston position corresponding to 50% and 100 % open ports, comparing the numerical results using four different large eddy simulation models: the one-equation model, the dynamic one-equation model, the localised dynamic one-equation model, and the mixed-scale model. The numerical models are implemented within the OpenFOAM platform and the results are compared with the experimental results.

We present the governing equations of the Navier-Stokes equation for incompressible flow, along with the equations of the four large eddy simulation models. The computational setup is then represented, and the numerical results are analysed and compared to the experimental results.

This part was published in the International Journal of Numerical Methods for Heat and Fluid Flow [8].

Part II: Mesh free and remeshed smoothed particle method

The chapter presents a Lagrangian, particle method for compressible flow. Starting by the key concepts of discretisation technique like finite element method, finite difference, finite volume, we go throughout the problems that those methods face, to present the need of mesh free methods.

Then we briefly presenting the mesh free method, and particle method. We show the limitation that the particle methods suffer from for specific application, to come to the point where we present our hybrid remeshed smoothed particle method (hrSPH).

Finally we present the governed equations and the algorithm for the hrSPH

method, followed by test benchmarks, and conclusion.

Part III: Brinkman penalisation technique coupled with the hybrid remeshed smooth particle method

In engineering application it is requir to simulate viscous flow around solid obstacles, we start presenting the two major methods that have been developed to deal with complex geometries problem. Body-fitted grid (BF) method], and immersed boundary (IB) methods, we show the limitation of the BF method, why the IB method was introduced as an alternative method for problems of fluid interaction around complex geometries, and the advantages of using it.

The Brinkman penalisation technique is an approach of immersed boundary method, where the boundary conditions are imposed by adding a penalisation terms to the momentum and continuity equations, and the obstacle is represented as a porous media. We represent the Brinkman penalisation for compressible flow, coupled with our hrSPH to simulate viscous flow with high Reynolds number. This part is closed with test benchmarks followed with a summery.

Chapter 2

Large eddy simulations of the influence of piston position on the swirling flow in a model two-stroke diesel engine

2.1 Background

The scavenging process in two-stroke marine diesel engines removes combustion gases from the engine cylinder and fills up the cylinder with the fresh air charge for the next cycle. Understanding the scavenging flow is crucial for the development of such engines, since it affects fuel consumption, engine cooling and production of pollutants [28]. In this work we consider the uniflow scavenging process where the exhaust valve is located at the cylinder head and the scavenge ports are located in the cylinder liner near the bottom dead centre (BDC). When the piston moves towards BDC, the exhaust valve opens while the piston slides to uncover the scavenge ports, thus initiating the scavenge process. The scavenge ports are angled such that the flow enters with an overall swirl in the

cylinder. In the succeeding compression stroke this swirling flow produces a pocket of swirling fresh air, which is important for the optimisation of the fuel spray mixing in the subsequent combustion phase.

Due to the rich fluid dynamic phenomena and many applications, there is a large body of work on flows with an overall swirl. Examples include the structure of helical vortices in swirling flow in a confined 'vortex chamber' at moderate Reynolds number [105], vortex breakdown [93, 73], vortex precession and the derived influence on combustors and dust separators [71].

Recently, [92] considered the effect of piston position on the scavenging and swirling flow in a simplified model of a large two-stroke marine diesel engine cylinder. Recognising the complexity of a real engine, [92] studied the flow in a "simplest possible" engine model: it contains no moving parts, and the set-up itself as well as the boundary conditions are rotationally symmetric, the exhaust valve is discarded, and the combustion is neglected. Of notable mention is the great care that was taken to impose an axisymmetric swirl by placing 60 thin guide vanes upstream of a contraction section in the inlet. By design, the set-up allows studies of fundamental aspects of the swirling flow in a uniflow scavenged engine. The effect of piston position was elucidated from time-averaged three-dimensional velocity fields of the air flow, obtained with stereo particle image velocimetry (SPIV) in the cylinder.

The current work is devoted to numerical modeling of the in-cylinder confined turbulent swirling flow of the experiment by [92]. We consider two cases of static piston position corresponding to 50 % and 100 % open scavenge ports, respectively. These cases yield qualitatively different regimes of the swirling flow, and complements previous studies that typically consider the case of fully open scavenge ports [21]. The experimental results will be compared with simulations using four different Large Eddy Simulation (LES) models: the one-equation model (OEM) cf. [104], the dynamic one-equation model (DOEM) cf. [59], the localised dynamic one-equation model (LDOEM) cf. [91], and the mixed-scale model (MSM) cf. [56]. The numerical models are implemented on the OpenFOAM platform. It is noted that swirling flows are notoriously difficult to simulate due to the inherent anisotropic turbulence. For example, Reynolds-Averaged Navier-Stokes (RANS) models are known to overestimate the rate of decay of swirl due to the eddy viscosity approach [100].

The remaining part of the chapter is organised as follows. Section 2.2 describes the numerical method and turbulence models, Section 2.3 describes the computational domain and boundary conditions, and Section 2.4 presents the results of the computations and compares them with measurements. Section 2.5 we discuss our results and give our conclusions.

2.2 Large eddy simulation

In order to model the confined turbulent swirling flow in [92] we make use of Large Eddy Simulations, where the flow is governed by the spatially filtered Navier-Stokes equations cf. [62].

$$\frac{\partial \bar{u}_j}{\partial x_j} = 0, \tag{2.1}$$

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial \bar{u}_i \bar{u}_j}{\partial x_j} = -\frac{1}{\rho}\frac{\partial \bar{p}}{\partial x_i} + \nu \frac{\partial^2 \bar{u}_i}{\partial x_j^2} + \frac{\partial \tau_{ij}}{\partial x_j}, \tag{2.2}$$

$$\tau_{ij} = \bar{u}_i \bar{u}_j - \overline{u_i u_j}, \tag{2.3}$$

Here, t denotes time, $\bar{u}_i$ the velocity component in the cartesian (x_i) coordinate directions, $\bar{p}$ the pressure, ρ the density, and ν is the kinematic viscosity. The resolved large scales ($\bar{u}_i$,$\bar{p}$) are explicitly computed on the computational grid, whereas small-scales fluctuations $(u_i - \bar{u}_i, p - \bar{p})$ are modeled through subgrid-scale stresses (τ_{ij}) cf. [44]

$$\tau_{ij} - \frac{1}{3}\delta_{ij}\tau_{kk} = -2\nu_\mathrm{t}\bar{S}_{ij}. \tag{2.4}$$

Here ν_t is the subgrid eddy viscosity, δ_{ij} is the Kronecker delta, and $\bar{S}_{ij}$ is the resolved strain stress tensor defined as

$$\bar{S}_{ij} = \frac{1}{2}\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right). \tag{2.5}$$

The trace of the subgrid scale stress (τ_{kk}) is usually included in the pressure $(\bar{p})$.

2.2.1 One-equation model

The One-Equation model (OEM) cf. [10] is based on a transport equation for the subgrid-scale turbulent kinetic energy (k)

$$\frac{\partial k}{\partial t} + \frac{\partial}{\partial x_j}(k\bar{u}_j) = \frac{\partial}{\partial x_j}\left[(\nu + \nu_\mathrm{t})\frac{\partial k}{\partial x_j}\right] + 2\nu_\mathrm{t}\bar{S}_{ij}\bar{S}_{ij} - C_\epsilon \frac{k^{\frac{3}{2}}}{\Delta}, \tag{2.6}$$

where

$$k = \frac{1}{2}\overline{u_i u_i} - \frac{1}{2}\bar{u}_i\bar{u}_i, \tag{2.7}$$

see also [104, 54, 110, 89]. Here, $\Delta = (\Delta_x \Delta_y \Delta_z)^{\frac{1}{3}}$, where Δ_x, Δ_y, and Δ_z denote the grid spacing in the x, y, and z directions, respectively. The constant

$C_\epsilon = 0.93$ is obtained from Kolmogorov's 5/3 law cf. [104]. The eddy viscosity is modeled as

$$\nu_t = Ck^{\frac{1}{2}}\Delta, \tag{2.8}$$

where the dimensionless parameter $C = 0.094$.

2.2.2 Dynamic one-equation models

The Dynamic One-Equation Eddy-viscosity Model (DOEM), allows the parameter C to vary in space and time during the simulation. The stress tensor is modeled as

$$\tau_{ij} - \frac{1}{3}\delta_{ij}\tau_{kk} = C\alpha_{ij}, \tag{2.9}$$

where α_{ij} is obtained from [44]

$$\alpha_{ij} = -2\Delta^2(2\bar{S}_{kl}\bar{S}_{kl})^{\frac{1}{2}}\bar{S}_{ij}. \tag{2.10}$$

The basic idea of the DOEM is to employ a second explicit filter ($\tilde{}$) with $\widetilde{\Delta} = 2\Delta$ to the filtered Navier-Stokes equations (Eqs. (2.1–2.3)), which yields the subtest-scale (STS) stress tensor

$$T_{ij} = \widetilde{\overline{u_i u_j}} - \tilde{\bar{u}}_i\tilde{\bar{u}}_j. \tag{2.11}$$

T_{ij} can be modeled as

$$T_{ij} - \frac{1}{3}\delta_{ij}T_{kk} = C\beta_{ij}, \tag{2.12}$$

where β_{ij} is

$$\beta_{ij} = -2\widetilde{\Delta}^2(2\tilde{\bar{S}}_{kl}\tilde{\bar{S}}_{kl})^{\frac{1}{2}}\tilde{\bar{S}}_{ij}, \tag{2.13}$$

The Germano identity [59] expresses the resolved turbulent stress L_{ij} as

$$L_{ij} = T_{ij} - \tau_{ij}, \tag{2.14}$$

and

$$L_{ij} - \frac{1}{3}\delta_{ij}L_{kk} = C(\beta_{ij} - \alpha_{ij}), \tag{2.15}$$

and an approximation for C is obtained by minimising in a least-squares sense the error

$$Q = (L_{ij} - \frac{1}{3}\delta_{ij}L_{kk} + CM_{ij})^2, \tag{2.16}$$

where $M_{ij} = \alpha_{ij} - \beta_{ij}$. The minimum error is obtained at $\frac{\partial Q}{\partial C} = 0$, with

$$C = -\frac{L_{ij}M_{ij}}{M_{kl}M_{kl}}. \tag{2.17}$$

The C obtained from Eq. (2.17) can attain both both positive and negative values. A negative value implies a locally negative eddy viscosity which may result in an unstable solution. To circumvent this, [59] proposed to average C across homogeneous directions. The standard DOEM model in OpenFOAM performs a simple global average, hence the C in the present DOEM model is only allowed to vary in time. The OpenFOAM implementation of the Localised Dynamic One-Equation Eddy-viscosity Model (LDOEM) proposed by [91] performs a local average using a simple top-hat filter.

2.2.3 Mixed scale model

The Mixed Scale Model (MSM) proposed by [56] and [84] is based on the velocity-vorticity formulation of the Navier-Stokes equations. Two spatial filters are used in the simulation, a filter denoted by $(\bar{})$ is used on the fine mesh, and a test filter denoted by $(\tilde{})$ on the coarse mesh. In Eq. (2.3) we defined τ_{ij} as the turbulent stresses from the spatial filtering, the turbulent stresses are modeled with an eddy viscosity as

$$\tau_{ij} = \nu_\mathrm{t} 2\bar{S}_{ij} - \frac{2}{3}k\delta_{ij}. \tag{2.18}$$

The eddy viscosity (ν_t) is determined by the mixed-scale turbulence model

$$\nu_\mathrm{t} = C\,|\bar{\omega}_i|^{\alpha}\, k^{(1-\alpha)/2}\Delta^{(1+\alpha)}. \tag{2.19}$$

Here, $\bar{\omega}_i = \partial_j \bar{u}_k - \partial_k \bar{u}_j$ is the vorticity, and $0 \leq \alpha \leq 1$ is a constant. The turbulent kinetic energy (k) is estimated from the test filter $(\tilde{})$ as follows

$$k = \frac{1}{2}\sum_{i=1}^{3}\left(u_i - \bar{u}_i\right)^2 \approx \frac{1}{2}\sum_{i=1}^{3}\left(\bar{u}_i - \tilde{\bar{u}}_i\right)^2, \tag{2.20}$$

Finally, from Eq. (2.19) we obtain the pure vorticity model if $\alpha = 1$

$$\nu_\mathrm{t} = C\,|\bar{\omega}|\,\Delta^2. \tag{2.21}$$

If $\alpha = 0$ we obtain the Bardina model [34]

$$\nu_\mathrm{t} = Ck^{\frac{1}{2}}\Delta \tag{2.22}$$

From a previous study [113] it was found that the model performs best when $\alpha = \frac{1}{2}$ such that

$$\nu_\mathrm{t} = C\,|\bar{\omega}|^{\frac{1}{2}}\, k^{\frac{1}{4}}\Delta^{\frac{3}{2}}. \tag{2.23}$$

2.3 Numerical set-up

The LES models are solved using the pisoFoam solver which is an OpenFOAM (version 1.6) build-in transient solver for incompressible and turbulent flows, where velocity and pressure are coupled using the PISO scheme [32]. We use the following numerical schemes: backward second order scheme for the temporal derivative and a second order central difference scheme for terms involving gradient, divergence and Laplacian operators. Finally, we make use of linear interpolation onto the subgrid-scale.

2.3.1 Computational domain

The computational domain is sketched in Fig. 2.1 and it consists of an inlet section, a cylinder, an exhaust pipe and an outlet. The cylinder radius is R and all other dimensions are shown in Fig. 2.1. Flow enters uniformly along a direction perpendicular to the cylinder axis, with an azimuthal velocity component, which ensures the overall in-cylinder swirling flow. The dimensions match the experiment by [92].

We simulate the flow on a grid with 8 million cells. The time step is chosen to ensure a stable and converged solution and to keep the Courant number in the range of $\overline{u}\delta t/\Delta_x < 1$ This is satisfied with a time step $\delta t V_\mathrm{b}/L_\mathrm{c} = 3.4 \cdot 10^{-4}$, where V_b is the bulk average flow speed in the cylinder and L_c is the length of the cylinder. Supplementary simulations on a 12-million node grid (not included here) show that the simulated flows are independent of the spatial discretisation.

2.3.2 Boundary conditions

At the inlet of the computational domain a uniform radial and tangential velocity is prescribed such that the flow enters with constant radial speed $V_{r,\mathrm{i}} = 0.23V_\mathrm{b}$ and constant tangential speed $V_{\theta,\mathrm{i}} = 0.11V_\mathrm{b}$, which is consistent with the 26° flow angle measured by [92]. At the outlet we require a zero velocity gradient. On the solid walls, full black lines in Fig. 2.1, a no-slip boundary condition is prescribed.

We have performed simulations with 0 %, 5 % and 10 % turbulence intensity (in the form of white noise) specified at the inlet, and found no impact on the results. For all simulations presented, a zero turbulence intensity is specified at the inlet.

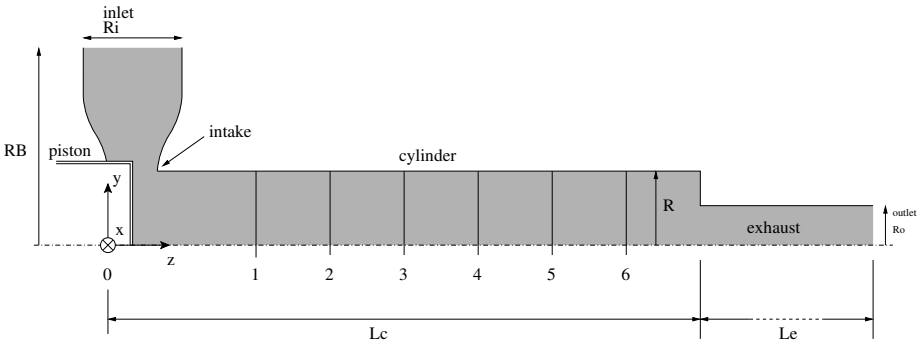


Figure 2.1: Sketch of the cylindrical computational domain shown in grey shades, in the case where the piston covers the intake by 50 %. Flow enters the domain through the horizontal inlet section and exits the domain through the vertical outlet. Notice that the exhaust is shortened in the figure. Data are extracted at the cross-sectional planes $z_{\{0-6\}}/R = \{0.00, 1.72, 2.77, 3.82, 4.87, 5.93, 6.98\}$.

2.4 Results

We study the effect of piston position by simulating the flow in two different geometries. One case corresponds to 100 % open intake (Section 2.4.1) and the other case corresponds to 50 % open intake (Section 2.4.2). We discuss in detail the mean axial and tangential velocity profiles obtained in the three cross-sectional planes $z_1/R = 1.72, z_3/R = 3.82, z_5/R = 5.93$ cf. Fig. 2.1, and compare the simulations at these positions with measurements by [92].

We characterise the flow with two dimensionless numbers. The Reynolds number

$$\mathrm{Re} = \frac{2RV_\mathrm{b}}{\nu}, \tag{2.24}$$

and the swirl number

$$\mathrm{S} = \frac{F_{\theta z}}{RF_{zz}} = \frac{\int_0^R \rho V_\theta V_z r \, \mathrm{d}r}{R\int_0^R \rho V_z^2 \, \mathrm{d}r}. \tag{2.25}$$

S expresses the ratio of the axial component of the angular momentum flux ($F_{\theta z}$) and axial component of the momentum flux (F_{zz}). A drawback of the definition of S is that it requires knowledge of the entire V_θ, V_z-fields in a given cross section, which is typically not known for experiments. In addition, S cannot be determined *a priori*—it is a consequence of the flow physics.

Instead, the swirl number is in this case approximated by the *design* swirl number

$$\mathrm{S} \approx \mathrm{S_d} = \frac{V_{\theta,\mathrm{i}}}{V_{r,\mathrm{i}}} \cdot \frac{2R}{4h}, \tag{2.26}$$

where $V_{\theta,\mathrm{i}}$ and $V_{r,\mathrm{i}}$ are specified at the inlet and h is the width of the inlet [92].

With the present boundary conditions and flow setting we have $\mathrm{Re} = 1.0 \cdot 10^5$ and $\mathrm{S_d} = 0.22$. In the experiment by [92] they had $\mathrm{S_d} = 0.23$ and their results were insensitive to a change from $\mathrm{Re} = 3.3 \cdot 10^4$ to $\mathrm{Re} = 6.5 \cdot 10^4$. Here we use of the measurements at $\mathrm{Re} = 6.5 \cdot 10^4$ to compare with the simulations.

Each simulation is run such that a non-dimensional time $T = tV_\mathrm{b}/L_\mathrm{c}$ of at least $T_2 = 30$ is covered. To avoid initial transients, we start averaging from $T_1 = 25$. We calculate the error associated with the mean velocity magnitude prediction from the expression [50]

$$\epsilon_{\langle u_i \rangle} = \sqrt{\frac{2\tau_\mathrm{i}}{T_2 - T_1}} \frac{\sigma_{u_i}}{\langle u_i \rangle}, \tag{2.27}$$

where τ_i is the integral time scale (minimum time between statistically independent samples), $\langle u_i \rangle, \sigma_{u_i}$ is the mean and standard deviation of the velocity component u_i. We estimate $\tau_\mathrm{i} = 0.035$ from the autocorrelation of a monitor point in the z_0-plane at $(x, y, z) = (0.27R, 0, 0.55R)$, for the 100 % open port case. Using Eq. (2.27) we obtain the error $\epsilon_{\bar{u}}$ of 10 % in the 100 % open port case and 4.6 % in the 50 % open port case.

Notice that in the following figures showing velocity profiles (Fig. 2.2 and Fig. 2.6), the SPIV measurements are translated in the x-direction such that the vortex centre (on average) coincides with $x = 0$. The measured off-axis position of the vortex centre is no larger than $0.05R$ and it is due to ever-present inaccuracies of the experiment. The uncertainty $\epsilon'_{\langle u \rangle} = \sigma_u/(\langle u \rangle \sqrt{N})$ associated with the measured average velocity component is of order 1.5 % (2.0 %) for the measurements with 100 % (50 %) open intake. In the formula, N is the number of independent samples.

2.4.1 100 % open intake

In this section we consider the case of fully open intake. We compare the simulated and measured mean velocity profiles, consider the spatial evolution of the swirl number, and comment on the three-dimensional mean velocity field.

In Fig. 2.2 the mean axial and tangential velocity profiles are shown for three streamwise positions z_1, z_3, z_5, cf. Fig. 2.1. The measured mean axial profiles

are symmetric with a velocity deficit in the central region, resulting in a wake-like profile. Furthermore, it is observed that the axial velocity profiles decrease radially in the range $|x/R| \gtrapprox 0.5$.

The measurements show plateaus with high axial velocity ($V_z/V_\mathrm{b} \approx 2.0$), and a central velocity deficit where the axial velocity is reduced to $V_z/V_\mathrm{b} = 0.16$, cf. Fig. 2.2*a*. It is seen that the DOEM and LDOEM are the only models capable of capturing the qualitatively the velocity deficit, LDOEM has the best quantitative agreement with the measurements. It should be noted that outside the vortex core region ($x/R \gtrapprox 0.2$) the four numerical models predict the same profile of V_z.

It is interesting to note that despite the disagreements between simulated and measured axial flow, except for the LDOEM model, there is a remarkable agreement in the predicted tangential velocity profile, as seen in Fig. 2.2*b*. The V_θ profile is well fitted by the Lamb-Oseen vortex profile $V_\theta = (\Omega c^2/x) \cdot (1 - \exp(-x^2/c^2))$ shown in green [24]. The Lamb-Oseen vortex is fitted to the LDOEM data using Ω and c as fitting parameters that minimise the residual in a root-mean-square sense.

In the downstream cross sectional plane shown in Fig. 2.2*c*, the axial velocity profile retains its wake-like profile although it is widened by diffusion and at z_5 (Fig. 2.2*e*) the axial profile is reduced to a plug flow. Note from Figs. 2.2*c,e* that at these positions all numerical models resolve qualitatively the wake-like axial profile, although they overpredict in the velocity magnitude at $x/R \lessapprox 0.4$. The measured and simulated tangential velocity profiles in these positions, agree well and as previously they are fitted well by appropriate Lamb-Oseen vortex profiles. It is worth noting that the significant viscous decay of the vortex profile observed in Fig. 2.2 is not reflected in the downstream evolution of the swirl number S (Eq. (2.25)). The streamwise evolution of S is plotted and it is observed that it evolves in a nonmonotonic fashion cf. Fig. 2.3. Specifically, S increases from S = 0.14 at z_1 to S = 0.19 at z_4. In attempting to elucidate the mechanism that gives rise to this behaviour it is noted that S expresses the ratio of fluxes of angular ($F_{\theta z}$) and axial (F_{zz}) momentum in the axial direction. Thus, S increases when the decay of F_{zz} is faster than the decay of $F_{\theta z}$. This is in fact the case as is shown in Fig. 2.4, where a relatively rapid decay of F_{zz} is observed in the range $1.0 < z/R < 3.0$. In the figure, the momentum fluxes have been normalised with the axial momentum carried by a plug-flow $F'_{zz} = \rho V_b^2 \pi R^3$. We note that while a change in $F_{\theta z}$ presumably is governed by viscosity, the change in F_{zz} is related to a change in the profile of the axial flow. Such changes can be caused by variations of the set-up's cross-section, or by the flow itself as e.g. vortex breakdown. In Fig. 2.4 we see that the decay of the axial flow profile, Figs. 2.2*a,c,e*, is accompanied by $F_{zz}/F'_{zz} \to 1$, as expected. In positions further downstream we expect F_{zz} to increase again due to the

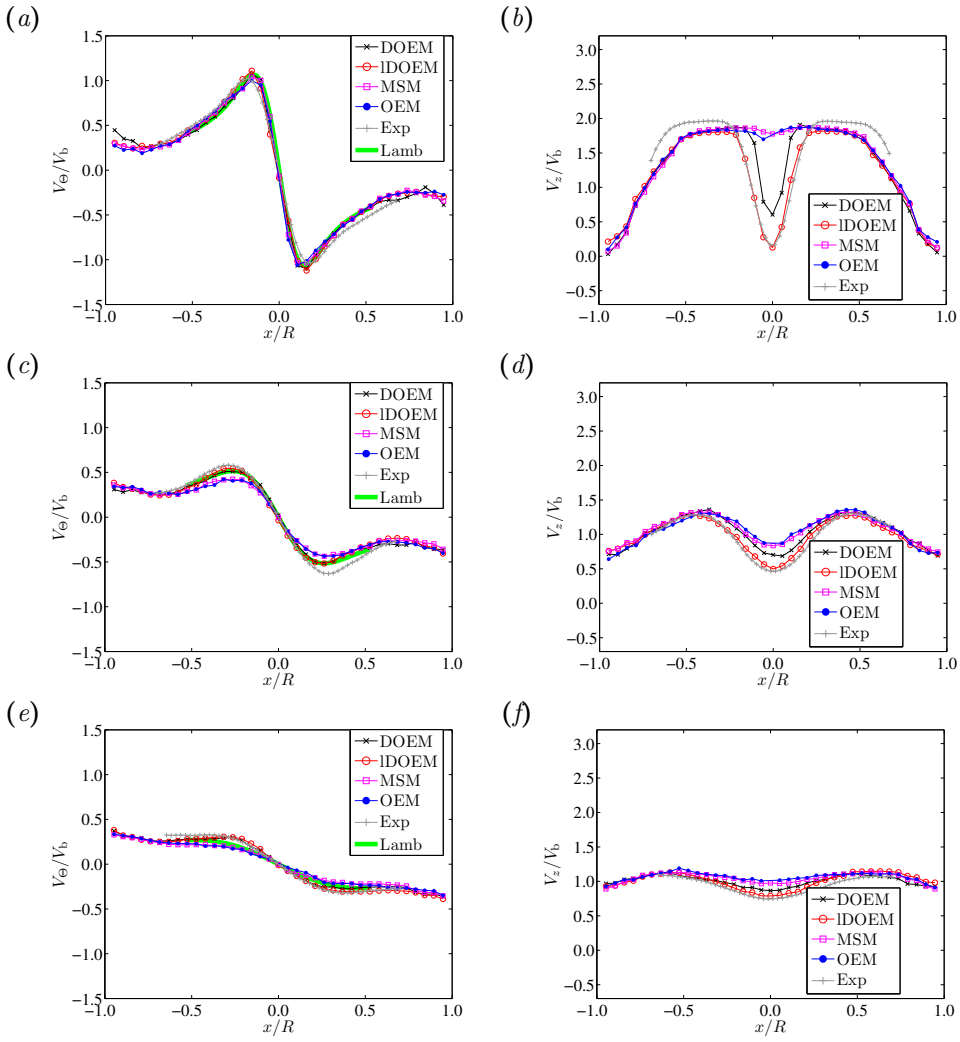


Figure 2.2: Time averaged velocity profiles for the 100 % open port case. The tangential (*a*,*c*,*e*) and axial (*b*,*d*,*f*) are obtained at the axial position (*a*,*b*) z_1; (*c*,*d*) z_3; (*e*,*f*) z_5. The abbreviations are 'DOEM' (Dynamic one-equation model), 'OEM' (One-equation model), 'LDOEM' (Localised Dynamic one-equation eddy-viscosity model), 'OEM' (One-equation model), 'MSM' (Mixed-scale model), 'Exp' (Measurements). The error associated with the measurements is of order 1.5 %.

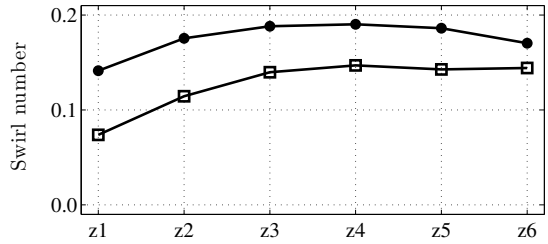


Figure 2.3: The swirl number decays along the computational mesh for the case of 100 % open intake (bullets) and case of 50 % open intake (open boxes). The exact coordinate of the abscissa labels are given in the caption of Fig. 2.1.

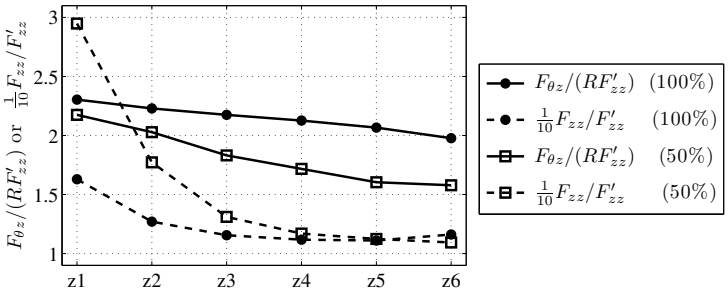


Figure 2.4: Momentum flux as function of the downstream position. Notice that the overall decay of angular momentum flux in the axial direction $F_{\theta z}$ is slower than the decay of axial momentum flux F_{zz}, which leads to the observed increase in swirl number, cf. Fig. 2.3.

upstream effect of the reduced exhaust diameter.

Figure 2.5 gives an impression of the three-dimensional mean flow for the 100 % open port case. Green and blue colours show streamlines that are released on two lines parallel with the z-axis in a meridional plane at the inlet $x = \pm 2.1R$. As observed in the inset, the streamlines move inwards along an almost straight line, before winding around the axisymmetric vortex core. Notice also from both figures the braid of red streamlines at the beginning of the main cylinder. The streamlines are released inside the recirculation zone and they effectively show the toroidal recirculation zone through which the other streamlines flow. Notice from the side view how the streamlines rotate approximately one-half revolution as they travel downstream in the main cylinder.

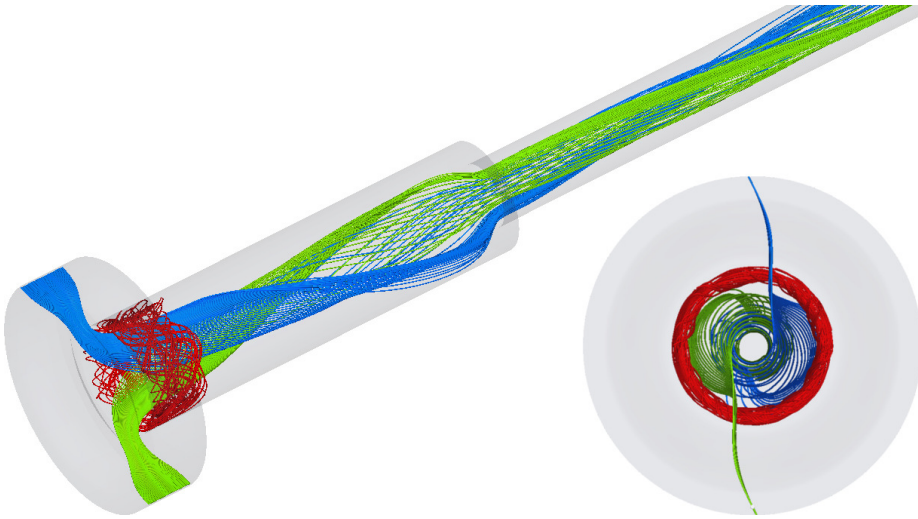


Figure 2.5: Visualisation of the three-dimensional flow by streamlines of the mean-field for the 100 % open port case. Notice the braid of red streamlines that shows the recirculation region at the upstream end of the main cylinder and the large pitch of the flow in the main cylinder. The lower right inset is a bottom-view that shows the swirling streamlines.

2.4.2 50 % open intake

We now consider a situation analogous to the piston covering half the intake, i.e. the computational domain is as shown in Fig. 2.1. The flow will be simulated using the LDOEM only since it showed to agree best with measurements in the 100% open intake case. The displaced piston presents a forward facing step to the flow, and flow separation now takes place at the sharp corner. Notice that Re is kept the same as for the 100 % open port case. From Fig. 2.6*a* it is apparent that displacing the piston has a dramatic effect on the mean flow profiles. In contrast to the case of fully open intake (Fig. 2.2) the mean axial flow now assumes an overall jet-like and symmetric profile in the measurements. The LDOEM predicts a profile in reasonable agreement with the measured profile. Notice that the axial flow is reversed in the region $|x|/R \gtrapprox 0.8$, which indicates that the recirculation bubble extends further downstream compared to the 100 % open intake case. We will elaborate further on this observation in section 2.4.3.

The simulated and measured mean tangential velocity profiles shown in Fig. 2.6*b*, display a poor agreement. The simulated flow has the overall tangential profile of a Lamb-Oseen vortex as shown by the green curve. In contrast, the measured profile is monotonically decreasing and it does not have a profile of a "standard" vortex. Upon comparison of the V_θ profiles of Fig. 2.2*b* and Fig. 2.6*b* it is interesting to note the difference in slope of the simulated tangential velocity profile near $x = 0$. For the 100 % open intake case, the slope of a linear fitting curve is -10.0 and for the 50 % open intake case the slope is -5.3, which shows that the maximum axial vorticity $\omega_z = (1/r)\partial(rV_\theta)/\partial r$ is reduced by 47 % in the case of 50 % open intake.

In the downstream position z_3, the axial flow persists having a jet-like profile cf. Fig. 2.6*c*, with a good qualitative agreement between the measured and simulated flows. The simulated profiles overestimate the measured profile by 10 % at $x = 0$. Notice from the tangential velocity profiles shown in Fig. 2.6*d* that it has the signature of solid-body rotation where in fact the thin wall boundary layer is not resolved by the shown data acquisition points. We estimate the boundary layer thickness from that of the flow past a flat plate $\delta/R \approx 0.37/(\mathrm{Re})^{-1/5}$ [27]. We obtain $\delta/R = 0.03$ which is closer to the wall than captured by the velocity monitor points shown in Fig. 2.6.

In the downstream position $z_5/R = 5.93$ the axial velocity profile has taken the form of a plug-flow with solid-body rotation, cf. Fig. 2.6*e,f*. Only a very narrow boundary layer exists near the cylinder wall. A similar profile was observed in the case of 100 % open intake, cf. Fig. 2.2*e*. The tangential profile remains solid-body like although the overall swirl decreases downstream due to viscosity. These observations are supported by the streamwise evolutions of the

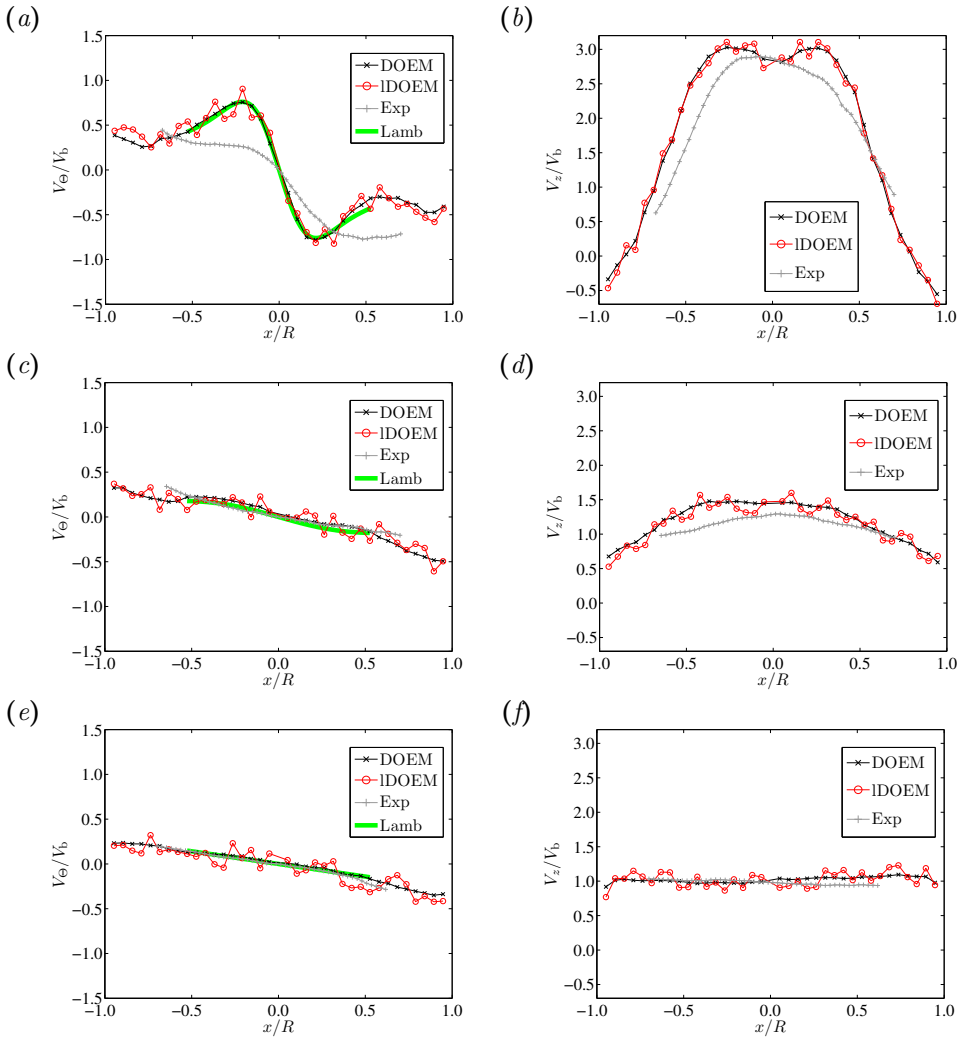


Figure 2.6: 50 % open intake case. (*a,c,e*) show the time averaged tangential velocity profiles and (*b,d,f*) show the time averaged axial velocity profiles. The axial positions are (*a,b*) z_1; (*c,d*) z_3; (*e,f*) z_5. The error associated with the measurements is of order 2.0 %.

momentum fluxes, cf. Fig. 2.4. Here it is seen that the jet-like profile yields a 80 % larger F_{zz} compared to the case of 100 % open intake. Again we observe a downstream decrease of F_{zz}. Notice that $F_{\theta z}$ is only slightly reduced, 5 % at z_1, compared with the 100 % open intake case. The succeeding decay of $F_{\theta z}$ is more rapid. We explain this observation with the stronger shear that yields an increased viscous decay of the angular momentum flux. In Fig. 2.3 we show the evolution of S as function of the downstream position (open boxes) for the present case and observe that, as expected, S assumes a smaller value than for the case of 100 % open intake.

The qualitative appearance of the streamlines (not shown) is similar to the 100 % open intake case (Fig. 2.5), i.e., a toroidal recirculation zone (of the average flow field) is located at the cylinder

wall immediately downstream of the inlet section. It is therefore worth noting that one cannot characterise the profile of the axial velocity profile from visual inspection of the mean velocity-based streamlines alone.

2.4.3 Flow structures

In Fig. 2.7 we show velocity magnitude fields in the meridional plane. Figure 2.7 shows the time-averaged velocity magnitude in the domain except for a part of the outlet for the case of 100 % open intake (Fig. 2.7*a*) and for the case of 50 % open intake (Fig. 2.7*c*). Notice here that the most significant difference in flow is observed immediately downstream of the inlet section. In both cases, the recirculation zone (on average) is located downstream of the corner where the flow turns from a predominantly radial to a predominantly axial direction. We estimate the streamwise extend of the separation zone from the sign of the axial velocity close to the cylinder wall and find that the streamwise length of the separation zone is $0.72R$ for the 100 % open intake case and $1.2R$ for the 50 % open intake case. The reason why the separation zone extends further downstream in the 50 % open intake case is the strong jet that is formed as flow enters the cylinder through the reduced intake. The qualitatively different nature of the axial flow profiles investigated in Fig. 2.2 and Fig. 2.6 is clearly recognisable in the medridional mean fields. Notice again that the velocity magnitude fields in the downstream third of the cylinder length are almost identical, as shown by the axial velocity profiles of Fig. 2.2*e* and Fig. 2.6*e*. A qualitative difference between the two mean velocity magnitude fields is the presence of two recirculation zones at the protruding piston. These are shown by selected streamlines in Fig. 2.7*e*. The panel shows a zoom of the region marked by a dotted line in Fig. 2.7*c*. We note that the shown streamlines are in fact three-dimensional streamlines projected onto the meridional plane. We

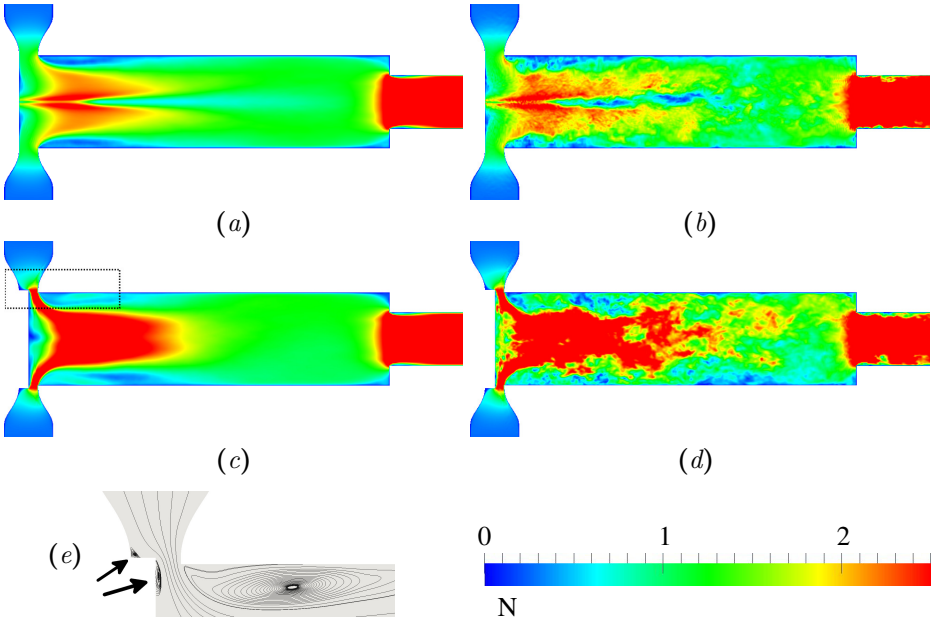


Figure 2.7: Meridional planes showing the velocity magnitude fields. (a,c) show the mean velocity magnitude for the cases of 100 % and 50 % open intake, respectively, and (b,d) show instantaneous velocity magnitudes for the same cases. In panel (e), selected stream lines show recirculation zones, two of which are shown with arrows, in the dotted region of panel (c).

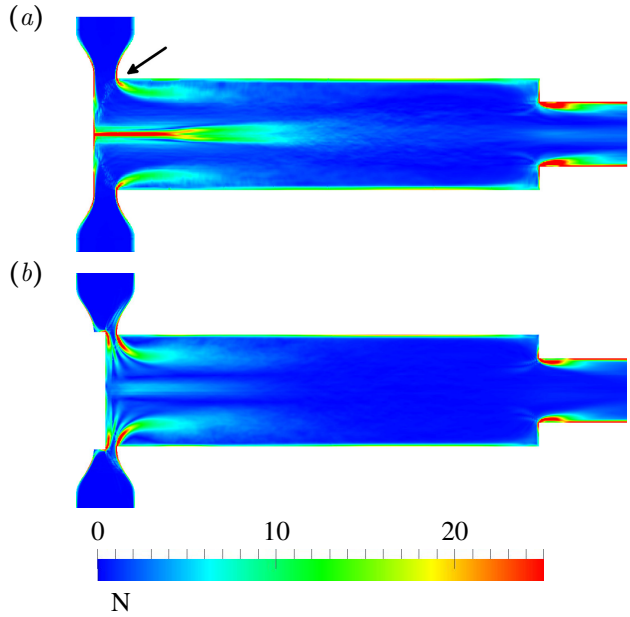


Figure 2.8: Meridional views showing the time averaged vorticity magnitude for the case of 100 % open intake (a) and the case of 50 % open intake (b).

observe a small recirculation zone before the step, marked by an arrow, and a larger recirculation zone on the piston surface. No recirculation zones were observed in the mean velocity magnitude field for the case of 100 % open intake.

In Fig. 2.7b,d we show typical examples of the instantaneous velocity magnitude field in the meridional plane, where the highly unsteady nature of the flow is evident, particularly in the 50 % open intake case. In the case of 100 % open intake we observe a symmetric velocity field, reminiscent of the mean field, whereas in the 50 % open intake case pronounced bursts of high-speed regions are seen. Presumably, these bursts appear due to interactions of shear layers created at the strong jets, which lead to large-scale disturbances of the flow.

Fig. 2.8 shows the time averaged vorticity magnitude in the meridional planes. We observe in the case of 100 % open intake (Fig. 2.8a), that a region of high vorticity magnitude exist in the centre of the cylinder effectively showing the compact vortex core. A moderate production of circulation is likewise observed at the sharp corner marked by an arrow. For the case of 50% open intake, (Fig. 2.8b) the vortex core is weaker is weaker, as discussed in Section 2.4.2, whereas a pronounced production of vorticity now takes place as the flow enters

the cylinder, as expected.

2.5 Summary

High Reynolds number swirling flows are generally hard to predict numerically—in particular RANS models over-predict the decay of swirl. In this contribution, we simulate the scavenging and swirling flow in the static geometry of a simplified model of a large diesel engine using LES with four different turbulence models. Upon comparison with experimental results, obtained with SPIV it is found that the both the "localised dynamic one-equation model" and the "dynamic one-equation model" [85, 91, 59, 62, 89] yield qualitatively and quantitatively good predictions of the mean axial and tangential velocity profiles. In contrast, the two alternative models, the "one-equation model" [104] and the "mixed-scale model" [56] both fail to predict qualitatively the correct profile of the axial flow profiles.

We show that the position of the piston has a dramatic effect on the measured and simulated flow fields: displacing the piston such that the intake is covered by 50 % (which corresponds to a 50 % blockage of the engine's scavenge ports) changes the time-averaged axial velocity profiles from a wake-like shape to a jet-like shape, and also diminishes the strength of the in-cylinder vortex. The partial intake closure also has more local consequences for the average velocity magnitude fields in the formation of two recirculation zones up- and downstream of the sharp corner. It is important to note that despite the difference in Reynolds number between the experiments by [92], good agreement between numerical and experimental results is observed.

We believe that our results carry two important messages. Firstly, the confined swirling flow can be predicted with a reasonable accuracy with LES using the dynamic one equation eddy model. Secondly, the piston position has a dominating influence on the in-cylinder flow, creating a qualitatively change in the flow topology.

2.6 Acknowledgements

We thank Valery L. Okulov and Dalibor Cavar for invaluable discussions and inputs. The research has been supported by the Danish Agency for Science Technology and Innovation (Grants No. 08-034075 and 09-070608), MAN Diesel

& Turbo, and the Danish Centre for Maritime Technology, We are grateful to ETH Zürich for providing computational resources.

Chapter 3

Meshfree and Particle methods

In general several steps should be applied to get a computational solution of any engineering problem, starting with a mathematical model that governs the physical problem, then the continuum is partitioned (discretised) into discrete elements. The most common discretisation techniques are: finite difference (FD), finite volume (FV), and finite element method (FEM). In computational engineering these method are popular and widely used. In the discretisation techniques the discretised element are connected via a topological map which is called mesh or grid, and an interpolation function is later built upon the mesh. However when using the discretisation techniques the physical continuum might be not always computable with the numerical scheme, and the interpolation may not produce a favourable results. Mesh distortion in Lagrangian computations, affect the accuracy of the computation, also in the cases of high gradient a much finer mesh is required therefor adaptively is required, to minimise the computation cost.

To handle the problems where time dependent geometry or boundary conditions are needed to model some problems, FEM uses adaptive remeshing technique to handle this type of problems. The remeshing technique, and mapping the variables state from the old mesh to the new one are not easy to handle. The mapping can introduce numerical errors, and remeshing should not be frequent.

In meshfree methods the continuum is discretise by a set of nodal point (particles) without any mesh [108].

Meshfree particle methods main advantage are:

- A direct link to physics and modeling as the particles represent extensive quantities.
- Relaxed stability limits because convection terms vanish in the Lagrangian description.
- Can handle large deformation easily, their connectivity can change in time , there is no need to generate a mesh, which is difficult in complex geometries.
- Universality: particle methods can simulate a wide range of models, including discrete and continuous ones as well as deterministic and stochastic ones.
- Can simulate cases where the partial differential equation (PDE) does not exist or has not been derived, by identifying particles with control volume and let them interact them according to the flow in between.
- Accuracy can be enhanced by and controlled easily, by adding more particles in time nn high gradient regions.
- Easy to represent complex geometries, as the discretised particles represent it, no need for difficult costly miss generation.

Particle methods can be classified in different way, based on physical principles they can be classified as deterministic and probabilistic. The majority of particle methods are based on probabilistic principles, like the Molecular Dynamics [60, 74], the Monte Carlo methods [1, 47], the Lagrangian Probability Density Functions (PDF) methods [13], and the Lattice Boltzman Method (LBM) [97].

Based on the computational formulation and the approximation of the partial differential equations, particle methods is classified as strong and weak formulation of the PDEs. The strong formulation includes Smooth Particle Hydrodynamics (SPH) [41, 40, 39, 11, 26], Vortex Method [36, 35, 5, 4, 25], and the Generalised Finite Difference Method [101].

Another class of particles method is based on different Galerkin weak formulations, those method are: Diffuse Element Method (DEM) [72], the Element Free Galerkin Method (EFGM) [14], the Reproducing Kernel Particle Method

(RKPM) [51], the h-p Cloud Method [57], the Partition of Unity Method [63, 31], and the Meshless Local Petrov-Glarkin Method (MLPG) [69].

In Particle methods the representative control volume are represented computationally as particles of finite volume, these particles are directly correspond to Lagrangian control volume. Particle methods connect closely to the physical or biological process of the simulated model, which makes the particle intuitive and easy to implement. As an example is the dynamic interaction of molecules and the non-equilibrium movement of atoms in nano-scales.

Computational particles are the base of the particle method, they are discretisation element, they don't necessarily correspond to real physical particles, rather a Lagrangian control volume.

Particles carry the following properties :

- Lagrangian position $x_p(t)$.
- Quantity (strength) $\omega_p(t)$.
- Volume $V_p(t)$.

Hence a particle is represented by the tuple of its attributes:

$$(x, \omega, V)_p, \tag{3.1}$$

where the comprehensive quantity ω_p contained in particle p is related to the underlying field u via the particle volume V_p as:

$$\omega_p = V_p u \tag{3.2}$$

In this dissertation we will mainly consider about Smooth Particle Hydrodynamics (SPH).

3.1 Smooth Particle Hydrodynamics (SPH)

Smoothed particle hydrodynamics method (SPH) was introduced by Gingold and Monaghan [3] and by Lucy [11] independently, with the aim to simulate astrophysical problem. The SPH particles movement is similar to the fluid flow,

and modelled by the classical Newtonian hydrodynamic equations. Representing the fluid as particles, which follow the fluid motion preserves the Galilean invariance, and due to the the nonlinear convection terms reduces the numerical diffusion of the flow.

Over the years SPH has been applied to many fields in science and engineering. Stam and Fiume [45] first used SPH to simulate fire and gas. Müller, Charypar, and Gross [64] developed a SPH method which can be applied on real-time fluid simulation, Hieber [23] used SPH for free surface flows and solid mechanics.

Turbulence modelling with SPH is rather a new field of research. Monghan [43] introduced a Lagrangian-averaged Navier Stokes turbulence model modifying the original SPH method showing a two-dimensional turbulence simulation. The method was computationally inefficient [38], largely owing to the iterative scheme required for velocity filtering. In 2007 three SPH turbulence models were introduced by Violeau and Issa [20] two algebraic and one based on the Reynolds stress model, Violeau and Issa simulated two-dimensional open channel turbulent flow and two-dimensional collapsing water column using their models, the results were satisfying comparing to the original SPH method, but poor if compared with grid-based method. Dalrymple and Rogers [2] used large eddy simulation (LES) turbulence model to simulate two-dimensional breaking waves with SPH. Robinson and Monaghan [66] used direct numerical simulation (DNS) with SPH and studied how will it performs for decaying in a no-slip square box, but their work is still limited to two-dimensional cases. Ellero [58] and Shi [112] studied cases with the high Mach number and isotropic homogeneous flows and showed that SPH in its original form has an effective implicit viscosity. Finally Adami [90] proposed a new algorithm combining the homogenisation of the particle configuration by a background pressure while at the same time reduces the artificial numerical dissipation.

3.2 Function approximation by particles

The physical problems in science and engineering are often described by continuous function subject to governing equations. These are descritised to be able to approximate the continuous fields function discreetly, and to give a good approximation of the differential operators. We need a particle function that approximate $u(x) : \mathbf{R}^d \to \mathbf{R}$ by particle.

The following step explains how that can be developed:

- **Integral representation** Writing the filed u as an integral with the help of the Dirac-delta identity

$$u(x) = \int u(y)\,\delta(x-y)\,dy, \tag{3.3}$$

where x is a location vector, assuming δ is a a function of particle location at position y. The problem with the integral is only exact for an infinite number of δ , The problem here is the integral means that the value u can be only recovered at particle locations and is unknown in between. But in practice we want to approximate the function with finite number of δ .

- **Regularisation** Next by replacing the delta function δ by smooth kernels of finite width ϵ so we get a smooth approximation which values is defined everywhere. δ is regularised as $W_\epsilon = \epsilon^{-d} W(\frac{x}{\epsilon})$ such that $\lim_{\epsilon\to 0} W_\epsilon = \delta$, with the condition $\int W dx = 1$. The pre-factor ϵ^{-d} is to rescale the function so the integral is always 1. We can think about the kernel W as a cloud of whatever extensive quantities the particle s carrying. this leads to

$$\langle u(x)\rangle = \int u(y)\,W(x-y,\epsilon)\,dy, \tag{3.4}$$

where x is a location vector, $u(y)$ is the interpolated function, W is a kernel function, that is used to localise the PDEs through the convoluted integral, ϵ is a scaling variable, and the symbol $\langle\,\rangle$ denotes the approximated interpolation value.

The more moments of delta function is conserved by the kernel, the more accurate the approximation is, if W conserve the first moments $r-1$ of δ we get the following order

$$u_\epsilon(x) = u_(x) + O(\epsilon^r) \tag{3.5}$$

This means,

$$\int x^s\,W(x)\,dx \neq \int x^s\,\delta(x)\,dx \quad \forall s \in 0...r-1 \tag{3.6}$$

So the moment of order $s=0$ has to be 1, and the higher order moments has to be 0. Non-negative kernels can never be of order higher than 2.

- **Discretisation** After building a smooth continuos function approximation on infinitely many particles, the next step is to discretise the approximation over a finite number of particles. Discretising the integral in

Eq. 3.4 using N-point quadrature with the particle locations as quadrature points:

$$u(x_m) = \sum_{p=1}^{N} u_p \, W(x_m - x_p, h), \tag{3.7}$$

where N is the number of particles, h is the distance between the particles. The function approximation error combined from: the discretisation (quadrature) error and the regularisation error:

$$u_\epsilon^h(x) = u_\epsilon(x) + O(\epsilon^r) + O(\frac{h}{\epsilon})^s. \tag{3.8}$$

s is the number of continuous derivatives of W. So the particles must overlap , the kernel widthϵ should be grater than the distance between the particles h, otherwise we will lose the smooth continuity and be back to step one.

In SPH the equations governing the flow are and expression of the particle-particle forces and flux interaction. PDEs operators are evaluated on the particles by converting them to equivalent integral operators, this conversion is not exact of course, so we end up approximating the operator for (mass, velocity, and energy) so they are algebraically conserved. The approximated operators are then discretised as a sum over all the particles.

Chapter 4

Three-dimensional remeshed smoothed particle hydrodynamics for the simulation of isotropic turbulence

4.1 Background

The Smoothed Particle Hydrodynamics (SPH) method was introduced independently by Gingold and Monaghan [3], and by Lucy [11], with the aim to simulate astrophysical problems. Over the years SPH has been extended and applied in many areas. Stam and Fiume [45] first used SPH to simulate fire. Müller et al. [64] developed an SPH method which can be applied on real-time fluid simulation. The SPH method was also extended in free surface flows problems [42], and low-Reynolds number viscous flows [29, 46, 83], Cummins et al. [37] extended SPH to simulate incompressible fluids, followed by Shao et al. [96] who propose an SPH simulation for Newtonian and non-Newtonian flows with a free surface. Cleary and Monaghan [107] extended SPH to heat transfer simula-

tion, and finally the method was developed for multi-phase flows simulation by Morris [82].

Turbulence modelling with SPH is a rather new field of research. Monaghan [43] introduced a Lagrangian-averaged Navier-Stokes turbulence model modifying the original SPH method for the simulation of two-dimensional turbulence. The method is computationally inefficient due to the reduced time step compared to the spectral method one [38], but the simulated energy spectrum and velocity profiles were found to be in good agreement with the results obtained using spectral methods.

Three SPH turbulence models were introduced by Violeau and Issa [20], two algebraic models, and one based on the Reynolds stress model. Two-dimensional open channel turbulent flow and two-dimensional collapsing water column cases were simulated, the kinetic energy, dissipation rate and eddy viscosity results were in good agreement with Monaghan [43] results, but the method may not be competitive in comparison with grid-based method, due to the small time step required resulting a large computational cost.

Dalrymple and Rogers [2] used a large eddy simulation (LES) turbulence model to simulate two-dimensional breaking waves with SPH. Robinson and Monaghan [66] studied how SPH performs in a direct numerical simulation (DNS) of decaying turbulence in a two-dimensional no-slip wall-bounded domain. They showed that the original SPH method can reproduce the energy cascade, which filled to an end state of a large monopole vortex that filled the domain, but their work was limited to two-dimensional cases.

Ellero et al. [58] and Shi [112] studied studied isotropic homogeneous turbulence cases with high Mach number and showed that SPH in its original form has an effective implicit viscosity. Finally Adami [90] proposed a new algorithm combining the homogenisation of the particle configuration by a background pressure which reduces the artificial numerical dissipation.

In this work we present a hybrid remeshed smoothed particle hydrodynamics method (hrSPH) for the simulation of three-dimensional turbulent flows. Rather than simplifying the framework and solve the system of equations solely on the grid, we combine an Eulerian mesh with Lagrangian particles to use the advantages of both schemes. We want to keep the free of the convection Courant Friedrichs Lewy (CFL) condition that the classical SPH method enjoys, whilst to take advantage of the computational efficiency to compute the derivatives on the grid, which is computationally cheaper than the nearest neighbour search of mesh free particle methods. The hrSPH also shares the adaptive character of SPH. The hrSPH framework is the first step toward a wider vision, in which the Lagrangian part (i.e. the particles) will play an important rule when a complex

geometry is needed. After all, it combines the abilities to apply Particle-Particle interactions to a part of the PDEs and Finite Differences to the other part.

The method is based on the remeshed smoothed particle method introduced by Chaniotis et al. [48] and Chatelain et al. [77]. The presented method differs from that the one of Chaniotis, as we take advantage of both the Lagrangian properties of the SPH along with the efficiency of Finite Difference scheme in which we interpolate the particles and compute the governing equations on the right hand side, rather than performing Particle-Particle interactions. We furthermore extend the framework of Chatelain et al. [77] in order to solve the full set of the Navier-Stokes equations. Subsequently, we add a the subgrid model to the system of equations. The particles are remeshed (uniformly reinitialised) onto uniform grid using a third order interpolation scheme to overcome the clustering or distortion of particles.

The mass and the impulse of the particles are interpolated onto the mesh, where the moments(mass, momentum, angular momentum, etc.) rate of change is computed. These are used to update the velocity and the position of the particles.

Direct numerical simulations (DNS), along with Smagorinsky [44] model are applied in this study. Details of the governing equations are presented next, followed with the hrSPH method, and finally the two- and three-dimensional results are presented.

4.2 Governing equations

The compressible flow is governed by the Navier-Stokes equations describing conservation of mass

$$\frac{D\rho}{Dt} = -\rho \frac{\partial u_i}{\partial x_i} \tag{4.1}$$

and conservation of momentum,

$$\rho \frac{Du_i}{Dt} = \frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j} + \frac{\partial \tau_{ij}^{sgs}}{\partial x_j} \tag{4.2}$$

where

$$\tau_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \delta_{ij} \frac{\partial u_k}{\partial x_k} \right), \tag{4.3}$$

where $\frac{D\diamond}{Dt} = \frac{\partial \diamond}{\partial t} + (u \cdot \bigtriangledown)(\diamond)$ denotes the material derivative, u_i is the velocity, p is the pressure, ρ is the density, τ_{ij} is the shear stress, μ is the dynamic viscosity, δ_{ij} is the Kronecker delta, and τ_{sgs} is the sub-grid stress tensor, which is zero in case of direct numerical simulation.

In the presented work the flow is uniquely described by the Reynolds number $Re = U\rho_0 L/\mu$, and the Mach number $Ma = U/c$. L is the characteristic length, ρ_0 is the reference density, U is the reference velocity, and c is the speed of sound.

To close the system (Eqs. (5.7-5.8)), the following equation of state is used,

$$p = \rho c^2 \tag{4.4}$$

4.2.1 Turbulence modelling

Direct numerical simulations are generally limited to low Reynolds number flow due to the available computational resources. In the present work we model the turbulent sub-grid stresses using the standard Smagorinsky model [44], defined as

$$\tau_{ij}^{sgs} = \rho \left(C_s \Delta\right)^2 \sqrt{2 S_{ij} S_{ij}} \widehat{S_{ij}} \tag{4.5}$$

with

$$\Delta = h, \tag{4.6}$$

where C_s is a non dimensional constant for which values ranging from 0.1 to 0.24 have been suggested in literature [95], Δ is the model length scale which is proportional to the the grid spacing Eq. (4.6), h are the mesh spacing, $\widehat{S_{ij}} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) - \frac{2}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}$ is the the filtered strain tensor, and $\left(C_s\Delta\right)^2 \sqrt{2 S_{ij} S_{ij}}$ is the norm of the filtered strain tensor, where $S_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$

4.3 Numerical modelling

4.3.1 The hybrid rSPH method

The basic idea of the hybrid remeshed smooth particle method (hrSPH) is to discretise the governing equations using Lagrangian particles carrying mass and impulse; the hybrid remeshed SPH method computes the right-hand side (RHS) of the governing equations by interpolating the mass and impulse of the particle onto a regular mesh, and from these, the flow density and velocity fields are obtained on the mesh nodes. These in turn allow for efficient calculation of the RHS using high order finite differences.

Several interpolating techniques have been developed, in the next section we are describing the interpolation function.

1. **Ordinary Interpolation**
   It is a second-order ordinary interpolation, conserving the zero-order moment and its first and second moment (impulse, and angular impulse). This interpolation Λ_2, is used in vortex method [81] is described as:

$$\Lambda_2(x,h) = \begin{cases} 1-s^2 & 0 \le s < \frac{1}{2}, \quad s = \frac{|x|}{h} \\ \frac{(1-s)(2-s)}{2} & 0 \le s < \frac{2}{3}, \\ 0 & s \ge \frac{3}{2}. \end{cases} \tag{4.7}$$

   where $|x|$ is distance of the particle to the mesh, h is the mesh spacing.

   So what to think about here is, the contribution of the old particle in the j^{th} row which carry properties Q_j will contributes in the new i^{th} particle at the $\widetilde{x}_i$ location with the interpolated quantities $\widetilde{Q}_i$ as in Fig. 4.1 [48]. The interpolation is obtained by tensorial product in each coordinates, and uses 3, 9, 27 point in one, two, and three dimensions,respectively. For example in two dimension is described as:

$$\Delta\,\widetilde{Q_i}(\widetilde{x}_i,\widetilde{y}_i) = Q_j(x_j,y_j)\,\Lambda_2(\widetilde{x}_i - x_j, h)\,\Lambda_2(\widetilde{y}_i - y_j, h) \tag{4.8}$$

   Must noted that Q_i must be extensive properties of the particle that is conserved.

$$Q_i = \begin{pmatrix} m_j \\ m_j u_j \\ m_j v_j \\ m_j w_j \end{pmatrix} \tag{4.9}$$

   Where m_j is the mass, $m_j u_j$, $m_j v_j$, and $m_j w_j$ are the three impulse form the momentum Eq. (5.8).

   It is noted form Eq. 4.10 that the interpolation function Λ_2 is discontinuous at the node locations, which results a larger error for the larger the interpolated quantities fluctuate.

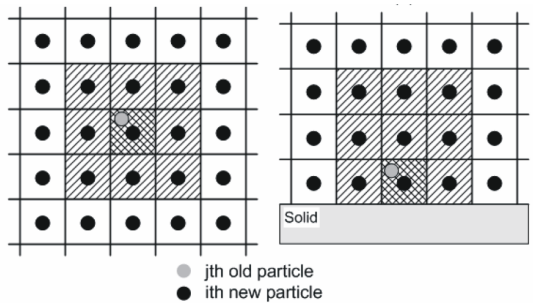


Figure 4.1: Detail of remeshing. The shaded cells are affected by the j^{th} particle. (left) Unbounded domain, (right) Bounded domain [48].

2. **Interpolation near solid boundaries**
biased on the ordinary interpolation, a second order interpolation function, that is conserving the same quantities is used for remeshing near the solid boundaries Fig. 4.1(right) [25]. As using the ordinary of the smoothing interpolation function near the solid boundaries extended the interpolation stencil to the interior of the solid, and hence can not be used. The interpolation stencil is described as:

$$\Lambda_2(x,h) = \begin{cases} 1 - \frac{3}{2}s + \frac{1}{2}s^2 & \text{1st cell away form the wall,} \quad s = \frac{|x|}{h} \\ s(2-s) & \text{2nd cell away form the wall,} \\ \frac{s(s-2)}{2} & \text{3rd cell away form the wall,} \\ 0 & \text{otherwise} \end{cases} \tag{4.10}$$

3. **Smoothing Interpolation**

The particle-to-mesh and mesh-to-particle interpolation is obtained using moment conserving interpolation. The interpolation was introduced to minimise the error that ordinary (not continuous everywhere) interpolations produce, through a moments-conserving interpolation (conservation of mass, momentum, angular momentum, etc.) [108, 79, 25]. However in mesh-to-particle interpolation, conservation of moments is generally not possible due to the non-uniform spacing of the target particles, though the interpolation error decreases as a power of the mesh spacing h. This power is called the order of convergence of the interpolation scheme.

The strengths (characteristics) of the particles (mass, and impulse) read:

$$w_p = \begin{pmatrix} m_p \\ m_p u_p \end{pmatrix}, \tag{4.11}$$

where, u_p is the three velocity component u, v, w, and m_p is the mass of the particle.

Fig. 4.2 shows 2-dimensional particle-to-mesh interpolation, where the strength of the particle is interpolated to the mesh using high order kernel as following:

$$\omega(x_m) = \sum_{p=1}^{N} \omega_p \, W(x_m - x_p, h), \tag{4.12}$$

where N is the number of particles, h is the mesh spacing, W is the high order kernel, ω_p is the strength of the particles, x_m is the position of mesh node m, and x_p is the position of particle p.

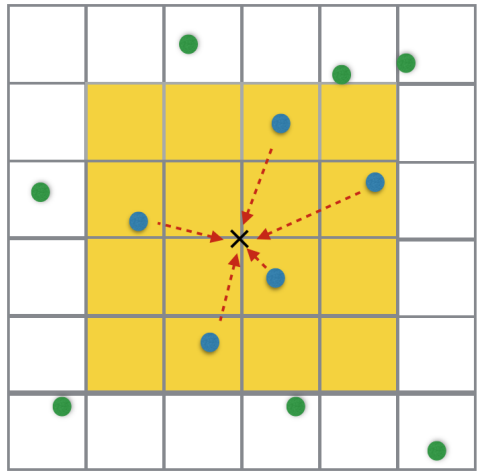


Figure 4.2: Schematic representation of particle-to-mesh interpolation in 2D using an interpolation function with support region $\pm 2h$ (shaded in yellow). Blue particles and are within the support region of the centre node (black) and hence assigned onto it. Green particles which lie outside the support region are not considered.

The smoothing interpolation is continuous everywhere in the interpolation stencil, and provide moment conserving interpolation. The $M'4$ introduced by [39] interpolate the strength of the particles to the mesh, the strengths are redistributed onto the surrounding mesh nodes as follows

$$M'4(x,h) = \begin{cases} 1 - \frac{5s^2}{2} + \frac{3s^3}{2} & 0 \leq s < 1, \quad s = \frac{|x|}{h} \\ \frac{(1-s)(2-s)^2}{2} & 1 \leq s < 2, \\ 0 & s \geq 2, \end{cases} \tag{4.13}$$

where $|x|$ is distance of the particle to the mesh.

This $M'4$ kernel has a four-point support with an error of $O(h^3)$, and the stencil of the discrete interpolation operator based on $M'4$ consists of 64 grid points.

The differential operator for the momentum equation are computed on the mesh, taking advantage of finite difference efficiency rather than using the particle-particle interaction. To maintain the Lagrangian advantages of the hrSPH , advection is taking part on the particles, by interpolating the rate-of-change in momentum to the particles (mesh-particle interpolation),

which is used to integrate the velocity and position of the particles forward in time.

We note the mesh controls the adaptivity and provides support for the fast evaluation of the pressure and stress tensor terms, the hrSPH mainly keeps the linear stability unconditional. The non-linear stability condition requires that particles trajectories do not cross [80]

$$\triangle t \leq C \left\| \bigtriangledown u \right\|_{\infty}^{-1}, \tag{4.14}$$

where $\triangle t$ is the time step, and C is the no linear Courant Friedrichs Lew (CFL) condition.

4.3.2 Remeshing

In the SPH method particles may cluster in one area of the computational domain and spread apart in another, as a result of the strain of the flow. When this occurs, the system looses the ability to recover continuous velocity and density fields.

When the distortion of the particle distribution occurres, the particle-mesh interpolation function is unable to ensure the continuity of the system, resulting in the inaccurate representation of the diffusion effect along with the pressure gradient (rate-of-change of momentum). To abrogate this problem, Chaniotis et al. [48] introduced the re-meshed smooth particle hydrodynamics method in which the position of the particles is periodically reinitialised to a uniform grid and the old particles properties are interpolated to the new ones. This interpolation has been implemented in several methods including particle methods [78, 81, 79].

By remeshing the particles using the high order interpolation kernel the following is accomplished:

- We retain the Lagrangian characteristic and the stability of the particle method. This gives us a large amount of control over the accuracy, as it leads to a more accurate computation of the derivatives compared to the classical particle-particle interaction (classical SPH).
- We decrease the computational costs thanks to:
  (a) The exploitation of the regularity of the particles.
  (b) Avoiding the costly nearest neighbour search. SPH method requires nearest neighbour searches for each particle to evaluate derivatives such as the pressure gradient. This adds to the computational costs of the classical SPH.
  We also remesh the particles and solve the right hand side of the equations on the grid. This increases the computational efficiency as nearest neighbours do not have to be found and the

speed of Finite Difference schemes is utilised. The cost of remeshing each time step is around 10% of the total costs [48] which is a small price compared to the total cost that the SPH required, along to the advantage of ensuring that the particles are always in-space, resulting an accurate approximation of the flow strain.

(c) The interpolation of the particle's characteristics on a uniform grid grants the method an increased computational efficiency to compute the derivatives using Finite Differences.

- Remeshing ensures that particles do not get too close to each other. This is an advantage as the pressure force in the momentum equation proportional to the derivative of the kernel, which in the case of $M'4$ is reduced to zero when the distance between the particles is small. In this case the pressure force becomes attractive resulting in significant errors in the classical SPH method [48].

The remeshing frequency in our framework can vary depending on the strain of the flow and the size of the time step. If the particles maintain their uniform distribution, as in the case of uniform flow fields without circulations and low Reynolds number (Re), remeshing only needs to be applied once per ten time steps, or even less frequent. In case of turbulent flows in which the flows recirculate however, remeshing is performed for every time step [48]. In the test cases in this framework we perform remeshing every time step, unless stated otherwise.

The accuracy of the method comes with a minimal additional computational cost while maintaining the adaptive character of the method. The implementation of high-order remeshing schemes improves the accuracy of hrSPH and additionally increases the computational efficiency of the algorithm.

The remeshing algorithm with finite support may result in numerical errors, as it may introduce substantial numerical diffusion. However, Koumoutsakos [79] and Chaniotis et al. [48] have shown that the introduced dissipation by remeshing and the errors of the computed gradients, induced by particle distortion, are proportional. These gradients remain substantially small if remeshing is performed at each time step. As discussed by Koumoutsakos [79] finally, remeshing acts like a subgrid scale and has a negligible effect on the accuracys

4.3.3 The hrSPH algorithm

Our method is divided into three parts:

(a) Computing the rate of change

i. Particle-mesh interpolation of the mass and impulse of the particle.

$$m(x_m) = \sum_{p=1}^{N} m_p \; W(x_m - x_p, h), \tag{4.15}$$

$$m(x_m)u(x_m) = \sum_{p=1}^{N} m_p u_p \; W(x_m - x_p, h) \tag{4.16}$$

where N is the number of particles, h is the mesh spacing, W is the high order kernel, up is the three velocity component u, v, w, and mp is the mass of the particle, x_m is the position of mesh node m, and x_p is the position of particle p.

ii. On the grid, obtain the velocity from the interpolated impulse

$$u(x_m) = \frac{m(x_m)u(x_m)}{m(x_m)} \tag{4.17}$$

iii. On the grid, compute the fluid density from the interpolated mass and the pressure from the equation of state Eq. (5.13).

$$\rho(x_m) = \frac{m(x_m)}{h^3} \tag{4.18}$$

iv. On the grid, compute the rate-of-change of the fluid momentum on the mesh (Δu_m) using finite-differences.

v. The rate-of-change of momentum is interpolated from the grid to the particles (Δu_p), Fig. 4.3.

$$\Delta u(x_p) = \sum_{p=1}^{N} \Delta u_m, W(x_m - x_p, h) \tag{4.19}$$

(b) Updating the particles
This part takes place on the set of particles, where the interpolated rate of change in velocity is used to update the velocity and position of the particles.

$$\vec{u}_p^{t+1} = \vec{u}_p^t + \Delta\vec{u}_p * \Delta t \tag{4.20}$$

$$\vec{x}_p^{t+1} = \vec{x}_p^t + \vec{u}_p * \Delta t \tag{4.21}$$

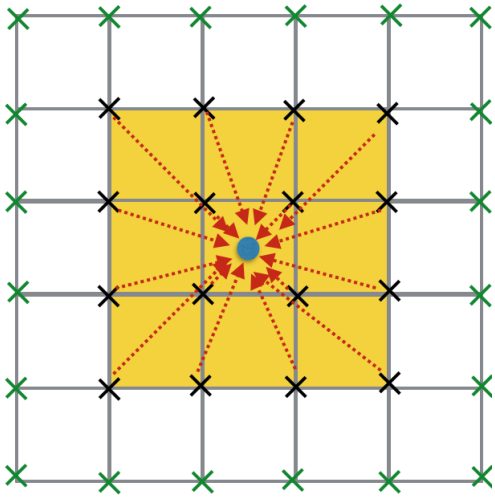


Figure 4.3: Schematic representation of mesh-to particle interpolation in 2D using an interpolation function with support region $\pm 2h$ (shaded in yellow). the blue particle and mesh nodes (black) are within the support region of the centre particle and hence assigned onto it. Green mesh and nodes lying outside the support and are not considered.

(c) Remeshing the particles:
In case of distortion and particle clustering (high CFL number, high gradients), interpolate the strengths of the particles to the mesh via $M'4$ interpolation function, generate a new set of the particles, interpolate the strengths back to the new set of particles.

For clarity an pseudo-code of the hrSPH algorithm follows,

Initialisation: Create particles carrying the initial mass m_p, and impulse $m_p u_p$;
while $t < endTime$ **do**
  On the particles:
  for *p=1 to N* **do**
    Interpolate the particle impulse and mass to the grid Eq. (4.15, 4.16).
  end
  On the grid:
  for *p=1 to N* **do**
    Obtain the velocity from the interpolated impulse Eq. (4.17);
    Obtain the density from the interpolated mass Eq. (4.18);
    Obtain the pressure from the equation of state Eq. (5.13);
    Right hand side computation:
    Using finite-difference, compute the rate-of-change of the fluid momentum Eq. (5.8);
  end
  for *p=1 to N* **do**
    Interpolate the the change of momentum to the particles Eq. (4.19).
  end

  On the particles:
  for *p=1 to N* **do**
    Update the velocity and position of the particles Eq. (4.20, 4.21).
  end
  if *Remeshing = true* **then**
    Do remeshing;
  end
end

Algorithm 1: The hrSPH algorithm

Solving the continuity equation Eq. (5.7) is not consistent with the system, rather the mass of the particles is updated via the $M'4$ function. This sequence is repeated in a third-order Runge-Kutta scheme [6] The pressure gradient is solved with a second order central difference scheme, while diffusion is computed using a second order central difference scheme. We want to note that the number of the particles is equivalent to the number of the grid points in all benchmarks in section (4), unless the number of the particles is stated explicitly. As mentioned before, remeshing is furthermore performed at every time step, unless stated otherwise.

4.4 Verification for the hrSPH method for viscous flow

To verify the method, we perform a series of benchmarks, including: two- and three-dimensional Taylor-Green flow [33], thin double shear layer [55], and three-dimensional isotropic turbulence. In this study the flow is characterised by the dimensionless Mach number Ma and the Reynolds number Re, which allow the reader to reproduce any of the benchmarks. The characteristic length scales L are the computational domain unless otherwise is stated, and the velocity u is normalised by either the maximum velocity or the reference velocity.

4.4.1 Two-Dimensional Taylor-Green flow

As a first test of the hrSPH method, we perform a simulation of the 2D incompressible Taylor-Green flow. Taylor-Green is a periodic flow of decaying vortices in the x-y plane as follows,

$$u(x,y,t) = -Ue^{bt}\cos\left(\frac{2\pi x}{L}\right)\sin\left(\frac{2\pi y}{L}\right) \tag{4.22}$$

$$v(x,y,t) = Ue^{bt}\sin\left(\frac{2\pi x}{L}\right)\cos\left(\frac{2\pi y}{L}\right) \tag{4.23}$$

$$p(x,y,t) = p_0 - \frac{U^2}{4}e^{bt}\left[\cos\left(\frac{4\pi x}{L}\right)+\cos\left(\frac{4\pi y}{L}\right)\right], \tag{4.24}$$

where $b = \frac{-8\pi^2}{Re}$, $Re = \rho_0 UL/\mu$ is the Reynolds number, L is the characteristic length of the system, ρ_0 is the reference density, μ is the viscosity. To approximate the incompressible reference solution, we choose a Mach number Ma is equal to 0.1, and the pressure reference is $p_0 = \frac{1}{M^2}$. The computational domain is $[L \times L]$ with periodic boundary conditions. We perform simulations for Reynolds numbers in the range $(10^0 - 10^3)$ to validate the accuracy of the method with the viscous effect (dominant, intermediate and minimal). The third order Runge-Kutta scheme is used throughout with a constant time step.

The flow maximum velocity decay behaviour with $Re = 10^0$ calculated using hrSPH with resolution of [64×64] is presented in Fig. 4.4 which shows a good agreement with the incompressible exact solution $U_{ex} = Ue^{bt}$.

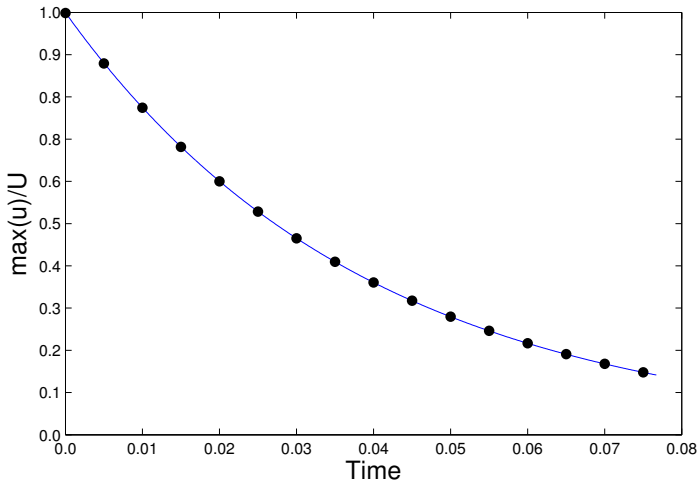


Figure 4.4: The maximum normalised velocity decay profile with $Re = 10^0$. Comparison of the hrSPH solution (-) with the exact incompressible solution(•). The hrSPH solutions shows a good agreement with the the incompressible exact solution $U_{ex} = Ue^{bt}$.

To test the accuracy of the method at higher Reynolds numbers we perform simulations at $Re = 10^2$. The predicted velocity decay shown in Fig. 4.5 is found in excellent agreement with the exact solution.

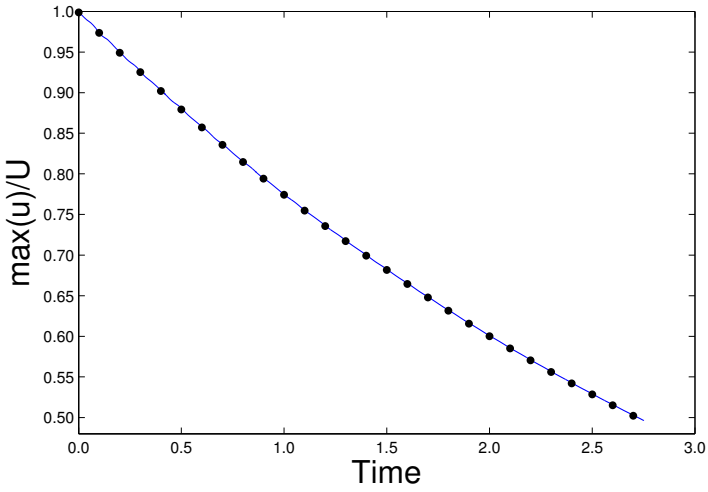


Figure 4.5: The maximum normalised velocity decay profile with $Re = 10^2$. Comparison of the hrSPH solution (-) with the exact incompressible solution (•). The hrSPH solutions shows a good agreement with the the incompressible exact solution $U_{ex} = Ue^{bt}$.

For the error analysis of the hrSPH simulation, the relative error (L_∞) is used

$$L_\infty(t) = \left| \frac{u(t) - U_{ex}(t)}{U_{ex}(t)} \right|, \tag{4.25}$$

where, $u(t)$ is the maximum velocity magnitude of the hrSPH simulation at time t, and $U_{ex}(t)$ denotes the maximum velocity magnitude of the exact solution at time t.

The relative error L_∞ for the hrSPH method calculation is between 0.5% and 1.4% for $Re = 10^2$ Fig. 4.6, which is twice as accurate then previously reported for SPH simulations [90, 48].

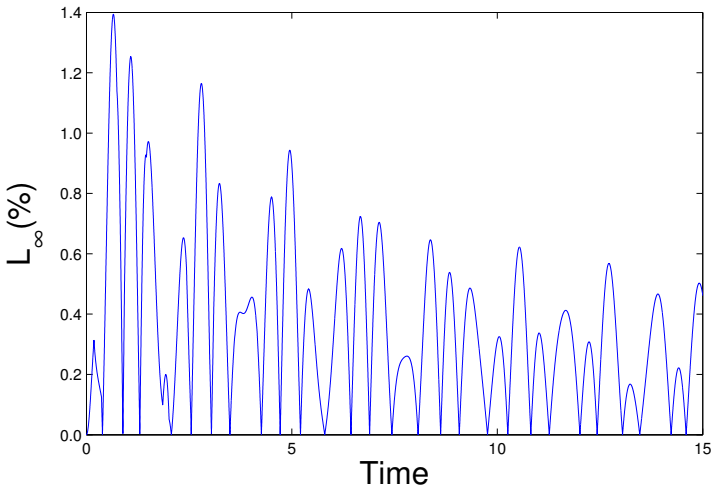


Figure 4.6: The relative error of the maximum velocity for the 2D Taylor-Green flow at $Re = 10^2$ using the hrSPH method with resolution of $[64 \times 64]$. The relative error L_∞ for the hrSPH method calculation is between 0.5% and 1.4% for $Re = 10^2$

The hrSPH relative error increases as the Reynolds number increases for a fixed grid resolution, with less numerical dissipation which is a plus advantage for the hrSPH method.

The maximum of the relative error $\max(L_\infty)$ shown in Fig. 4.7, for a 64×64 resolution is less than 2% for Re in the range considered $1 - 10^3$.

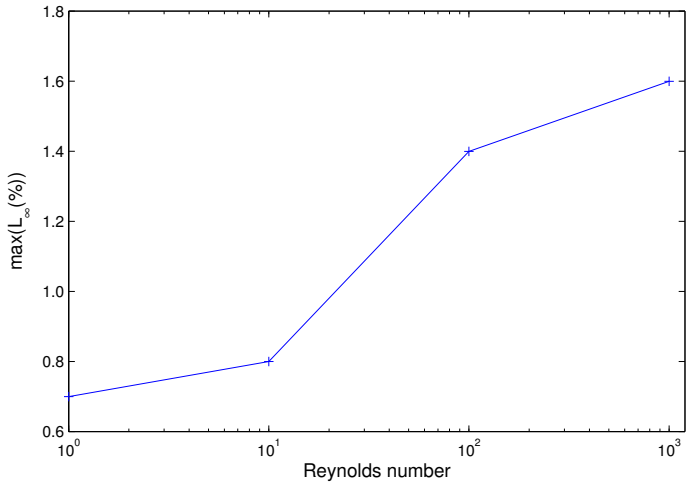


Figure 4.7: $\max(L_\infty)$ error of the hrSPH simulations of the 2D Taylor-Green flow for different Reynolds number with a fixed grid resolution $[64 \times 64]$. The maximum of the relative error $\max(L_\infty)$ is less than 2% for Re in the range considered $1 - 10^3$.

Finally we tested the convergence rate of the relative error L_∞ for a spatial grid refinement. The profile of the maximum relative error $\max(L_\infty)$ of the hrSPH simulation with different resolution $[16 \times 16, 32 \times 32, 64 \times 64$, 128×128, and $256 \times 256]$ is presented in Fig. 4.8

The hrSPH exhibits a third order convergence in space, with third order diffusion and pressure gradient, which is consistent with the order of the $M'4$ interpolation function as represented in Fig. 4.8.

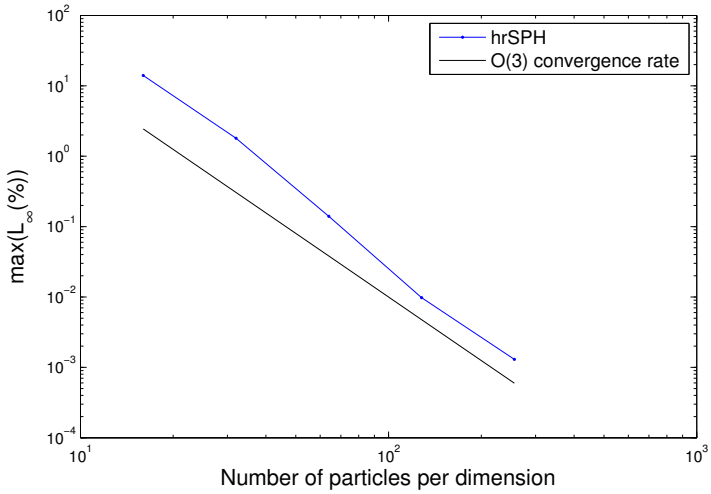


Figure 4.8: $\max(L_\infty)$ error of the hrSPH simulations for the Taylor-Green flow with different resolutions with $Re = 100(\cdot)$, along with the third order convergence rate. The hrSPH exhibits a third order convergence in space, which is consistent with the order of the $M'4$ interpolation function.

4.4.2 Thin double shear layer

To illustrate the performance of the hrSPH method on under resolved flow, we simulated the evolution of a thin double shear layer. The thin double shear layer, which is often considered to be too difficult to simulate due to the produced small scales. The main challenge of the this problem as showed by Brown and Minion [55], occurs when the method is producing the spurious structures, in the case when the flow is sufficiently under-resolved. Brown and Minion [55] tested several numerical schemes, and showed that given a sufficient resolution (256×256) all the numerical schemes provided a reasonably accurate solutions. But given a coarser mesh (128×128) the methods generate a non physical spurious vortex in the shear layer between the two vortices, with an early oscillations at $t = 1.0$.

Drikakis and Smolarkiewicz [17] studied the spurious structure, aiming to understand the numerical mechanism behind it. They indicated that the generation of the spurious structure depends on the choice of the advective scheme.

The computational domain is a unit square with periodic boundary conditions. The flow velocity $\mathbf{u} = (u, v)$ is initially consists of a horizontal

shear layer of a finite thickness as

$$u(x,y) = \tanh(80 \times \min(y - 0.25, 0.75 - y)) \quad (4.26)$$
$$v(x,y) = \delta \sin(2\pi(x + 0.25)), \quad (4.27)$$

In the simulation we set $\delta = 0.05$, Reynolds number $Re = 10^4$, and Mach number $Ma = 0.1$, initially we start with a uniform pressure and density. Fig. 4.10 shows the evolution of the vorticity for three different mesh resolutions; a fine one with 400×400, a 200×200, and a coarser one with 100×100. We note that using the hrSPH method, with a relatively low resolutions we are able to overcome the development of the spurious vortex compared to previous studies [19] cf. Fig. 4.9.

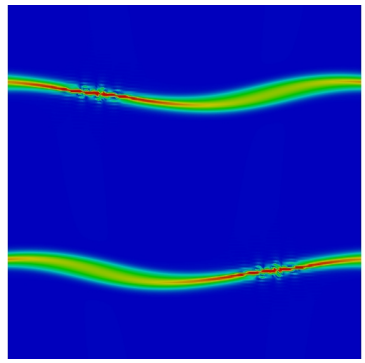


Figure 4.9: The development of a spurious vortex for the lower resolutions 256×256

However with the coarse 100×100 mesh the hrSPH method produces the spurious structure and the simulation failed, which agrees with the previous studies in [55].

The vorticity evolution for both resolutions 400×400, and 200×200 is presented in Fig. 4.10, both cases were able to avoid the spurious structure.

With Reynolds number $Re = 3 \times 10^3$ a resolution of 100×100 the hrSPH is able to simulate without producing the spurious structure. At time $t \approx 4$ we notice that oscillations are produced, as mentioned by Minion [55], however this problem occurred after a long simulation time, as shown in Fig. 4.11.

As the hrSPH enjoys the benefits of a Lagrangian advection, the method is able to provide accurate results for the thin double shear layer with a lower mesh resolution meshes compared to previous studies [55, 17, 19].

Finally the calculated maximum error of the relative effective viscosity [12] $\mu_{eff}(t) = \frac{\epsilon(t)}{\varepsilon(t)}$ to the physical viscosity μ, where ϵ is the dissipation rate Eq. (4.28), and ε is the enstrophy Eq. (4.29).

$$\epsilon = \frac{dE_k}{dt} \tag{4.28}$$

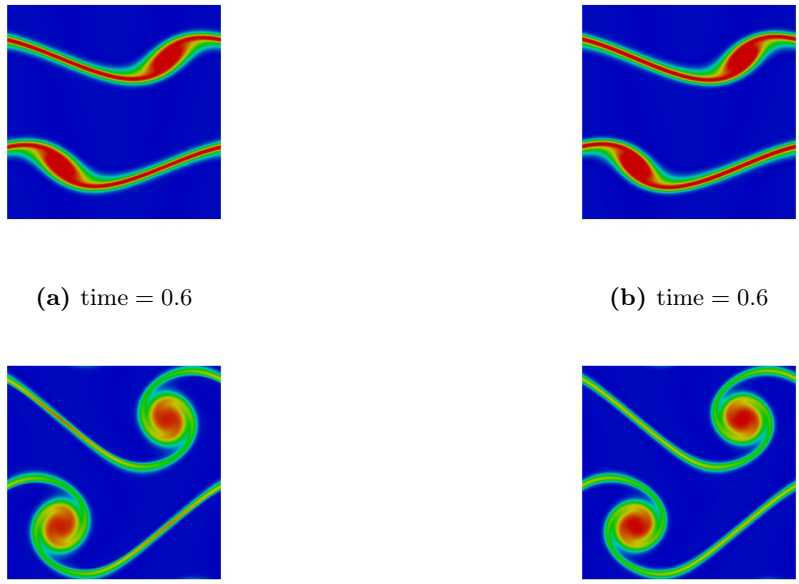
$$\varepsilon = \int_\Omega \frac{\omega \cdot \omega}{2} d\,\Omega, \tag{4.29}$$

where E_k is the kinetic energy

$$E_k = \int_\Omega \rho \frac{u \cdot u}{2}\, d\Omega, \tag{4.30}$$

and ω is the vorticity, and Ω is the computational domain.

For the thin double shear layer flow with $Re = 10^4$ and resolution of 200×200 at $t = 1$, the maximum error of the relative effective viscosity is equal to 2%.



(a) time = 0.6 **(b)** time = 0.6

(c) time = 1.0 **(d)** time = 1.0

Figure 4.10: Vorticity magnitude of the thin double shear layer simulation with $Re = 10^4$. Left column using 200×200 particles, and right column using 400×400 particles. The hrSPH is able to avoid the development of the spurious structure with a lower grid resolution compared to [19].

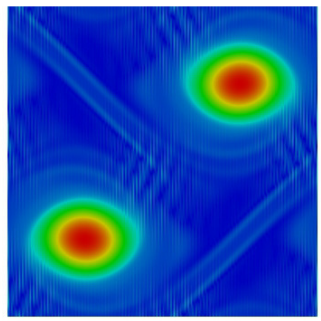


Figure 4.11: The produced oscillation for the thin double shear layer using the hrSPH using a coarse 100×100 mesh. The hrSPH is able to avoid the development of the spurious structure, however oscillations are produced at time $t \approx 4$.

4.4.3 Three-dimensional Taylor-Green flow

This benchmark considers direct numerical simulation (DNS) with hrSPH method, three-dimensional Taylor-Green is a periodic flow of decaying vortices in the $x - y - z$ plane, with the following initial conditions,

$$u(x,y,z) = U \sin\left(\frac{2\pi x}{L}\right)\cos\left(\frac{2\pi y}{L}\right)\cos\left(\frac{2\pi z}{L}\right) \tag{4.31}$$

$$v(x,y,z) = -U \cos\left(\frac{2\pi x}{L}\right)\sin\left(\frac{2\pi y}{L}\right)\cos\left(\frac{2\pi z}{L}\right) \tag{4.32}$$

$$w(x,y,z) = 0 \tag{4.33}$$

$$p(x,y,z) = p_0 + \frac{\rho_0 U^2}{16}\left(\cos\left(\frac{2\pi x}{L}\right) + \cos\left(\frac{2\pi y}{L}\right)\right)\left(\cos\left(\frac{2\pi z}{L}\right) + 2\right), \tag{4.34}$$

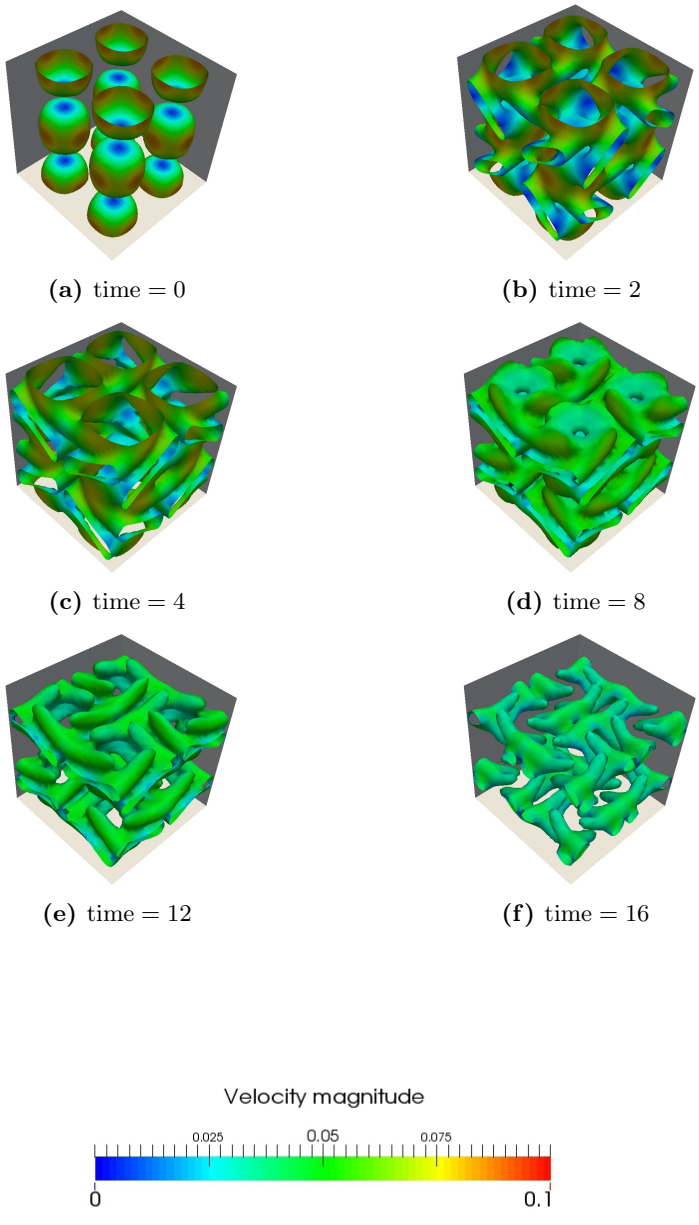
where, U_0 is the reference velocity, the Mach number $Ma = 0.1$, L is the respective length, and p_0 is the reference pressure, which is determined from the reference density ρ_0 by the equation of state Eq. (5.13).

The aim of this test case is to test the accuracy of hrSPH for three-dimensional viscous flow with Reynolds number $Re = 1600$, using direct numerical simulation. The flow is confined in a cube with periodic boundary conditions defined as $0 \leq x, y, z \leq 2\pi$. The computational meshes are regular cartesian grids of $64^3, 128^3, 256^3$ resulting in $\Delta x = 0.01, 0.05, 0.025$.

The third order Runge-Kutta is used for time interpolation. Fig. 4.12, represents the isosurface of the vorticity magnitude at different times. The evolution of the kinetic energy over time is presented in Fig. 4.13 (a), we observe that the hrSPH method is capable to capture the basic dynamic flows for different grid resolutions and is in a good agreement with the reference solution [99]. The change in the kinetic energy over time for the three grid resolutions is insignificant, however the close up is shown in Fig. 4.13 (b), shows that the coarser grid contains less energy than the finer one, and the energy decays faster as time evolve. At early time steps as the vortices begin to evolve and maintain their shape, this phase lasts approximately until $t = 7$ where the smooth structures begins to suffer changes in their structure as the flow becomes turbulent, at $t = 9$ the coherent structure breaks down. Fig. 4.14 depicts the evolution of the dissipation rate (ϵ) Eq. (4.28), and the enstrophy (ε) Eq. (4.29). Fig. 4.14(a) shows the time history of the enstrophy, it is clear that there is a large change in the peak dissipation rate for the coarser grid ($\Delta x = 0.01$), this peak is improved by increasing the grid resolution until we reach a good agreement with the reference solution at grid resolution $\Delta x = 0.025$. The dissipation rate is represented in Fig. 4.14(b), we examine a large difference in the dissipation peak at $t = 9$ where the coarser grid fails to estimate the correct dissipation peak. The finer grid, with $\Delta x = 0.025$, is consistent with the reference solution [106]. We calculated the error of the relative effective viscosity $\mu_{eff}(t) = \epsilon(t)/\varepsilon(t)$ to the physical viscosity. The maximum error in the relative effective viscosity is about 2% for the hrSPH method.

To finally test the effect of the remeshing frequency, we run the same test case at $Re = 1600$ with $\Delta x = 0.05$, whilst decreasing the time step by a factor of two ($\Delta t = \Delta t/2$) and keeping the remeshing frequency the same, we used the solution of the kinetic energy evolution with $\Delta x = 0.05$ presented in Fig. 4.13 (a) as a reference solution to calculate the relative error $L_\infty(\%)$. The reason is to check if our computation is well converged in time resolution. The results in Fig. 4.15(b) show that no effect on the solution and the relative error $L_\infty(\%)$ is approximately 0%. This leads us to conclude that any changes in the solution will be a result of changing the remeshing frequency. We thus also performed two test cases in which the remeshing frequency is increased by a factor of two and four respectively, while decreasing the time step with a factor of two and four as well. The results in Fig. 4.15(a), (b) show that the remeshing frequency affects the results when the flow contains a substantial kinetic energy and small scales. The relative error of the kinetic energy is approximately 2% when remeshing is performed at every time step and approximately 4.5% if remeshing is performed once per four time steps. This can be explained based on the fact that remeshing allow the system to regain it's regularity

as it ensures the particles' uniform distribution, even when flows with a large Reynolds number are considered. As the kinetic energy decreases as a function of time, the relative error reduces to well below 1%.

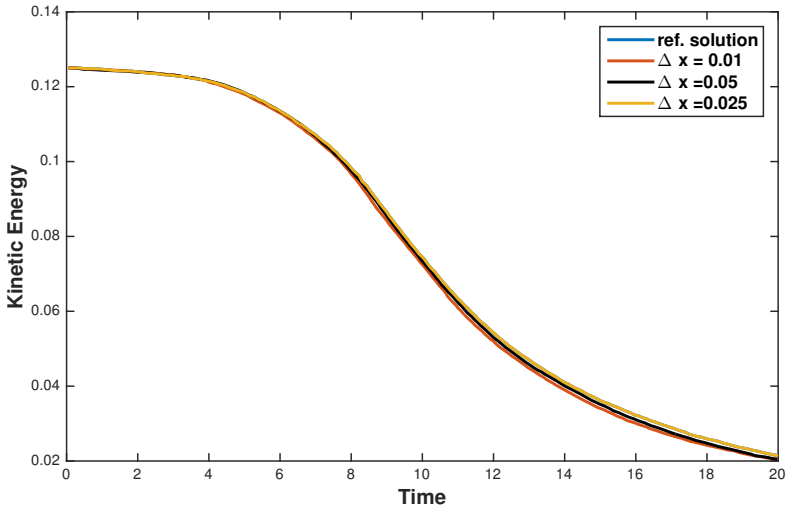


(a) time = 0 **(b)** time = 2

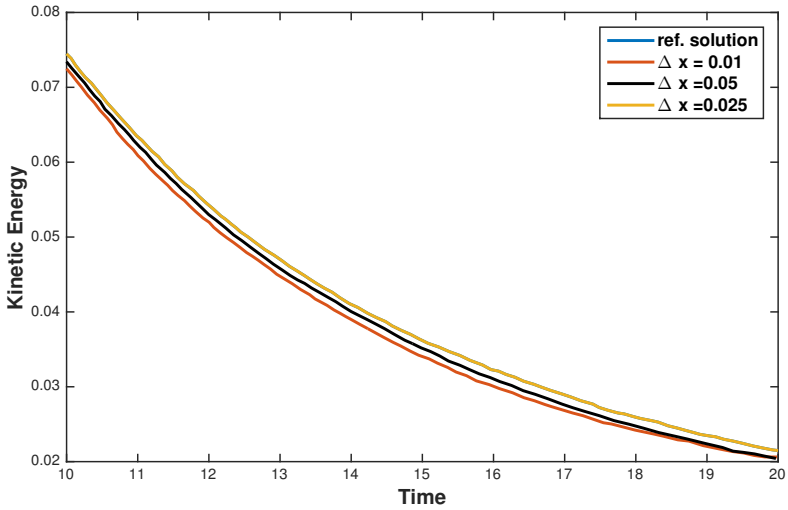
(c) time = 4 **(d)** time = 8

(e) time = 12 **(f)** time = 16

Figure 4.12: Isosurface of the vorticity magnitude for the Taylor-Green simulation at $Re = 1600$ with $\Delta x = 0.025$.

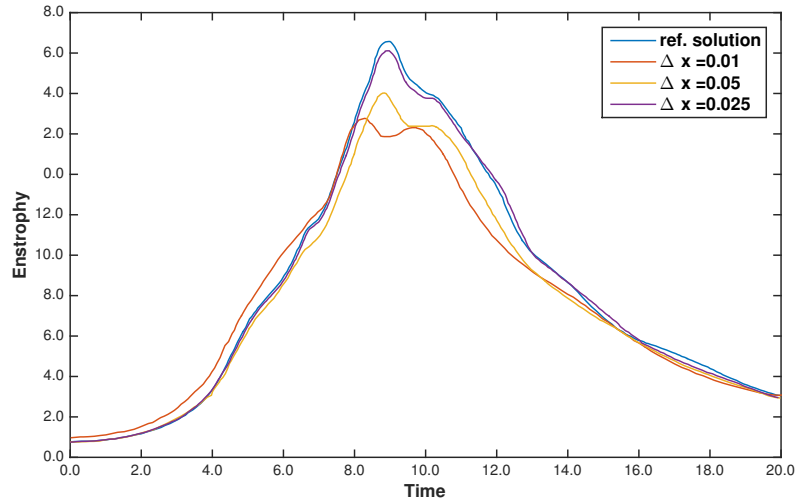


(a) Complete simulation.

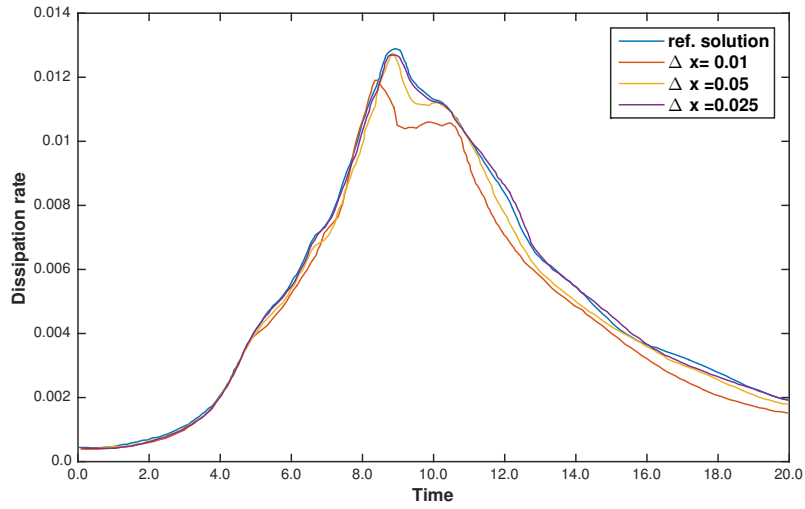


(b) Part of the simulation 10<time<20.

Figure 4.13: Evolution of the kinetic energy for the 3D Taylor-Green simulation, $Re = 1600$, using the hrSPH method, with different resolution along with the reference solution. In (a), we observe that the hrSPH method is capable to capture the basic dynamic flows for different grid resolutions and is in a good agreement with the reference solution. The close up in (b), shows that the coarser grid contains less energy than the finer one, and the energy decays faster as time evolve.

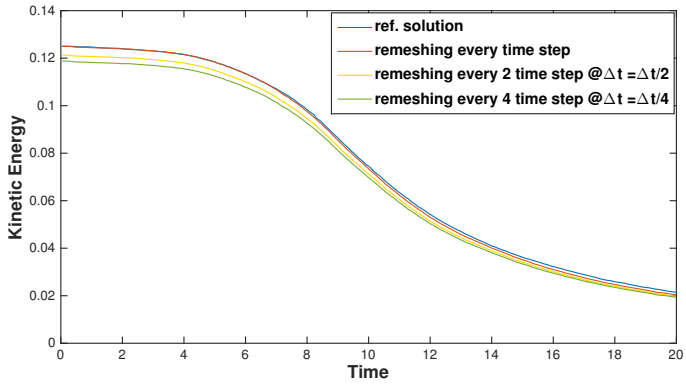


(a)

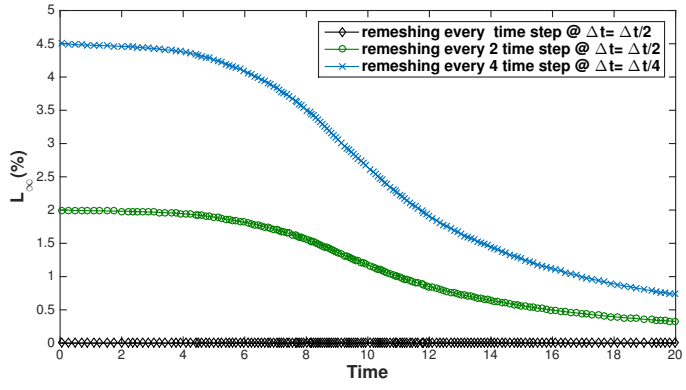


(b)

Figure 4.14: Evolution of enstrophy (a), and dissipation rate (b) for the simulation of the 3D Taylor-Green at $Re = 1600$ with different resolution using the hrSPH method. In (a) there is a large change in the peak dissipation rate for the coarser grid, this peak is improved by increasing the grid resolution until we reach a good agreement with the reference solution. In (b) we examine a large difference in the dissipation peak at $t = 9$where the coarser grid fails to estimate the correct dissipation peak. The finer grid is consistent with the reference solution.



(a)



(b)

Figure 4.15: The effect of remeshing frequency on the the evolution of the kinetic energy for the 3D Taylor-Green simulation (a), and the relative error of remeshing every 2 and 4 time steps (b), for the simulation of the 3D Taylor-Green at $Re = 1600$.

4.4.4 Three-dimensional isotropic turbulence

We use hrSPH method to simulate three-dimensional isotropic turbulence in a periodic cube of size $L = 2\pi$ with a resolution of 64^3. The initial conditions are obtained from the JHU Turbulence Database Cluster [111] a 1024^4 space-time history of a direct numerical simulation of incompressible isotropic forced turbulent flow at $Re \approx 1460$.

The data from the database contains the three velocity components and the pressure, the data is for incompressible flow. A uniform non-dimensionalised

pressure $p^* = \frac{p}{\rho U^2} + 1$ is added to the database pressure, with Mach number $Ma = 0.1$. The 1024^3 resolution mesh is filtered with a Gaussian filter with a specified cutoff to reduce the noise, and then down sampled to the desired resolution (64^3).

Both the DNS and Smagorinsky models are used to predict the three-dimensional isotropic turbulence problem, the Reynolds number ranging from $Re = 300$ to 10^4. The aim here is to study the Reynolds number threshold for both models by maintaining the same resolution.

The simulation results that the DNS fails to properly predict the turbulent flow for $Re > 2 \times 10^3$. This is caused by the insufficiently fine mesh resolution required to solve the many small scales the DNS is taking into account. The Smagorinsky model on the other hand fails with the same resolution for $Re > 6 \times 10^3$, with the same resolution (64^3).

The energy spectrum is calculated as following, for each component of the velocity fields on the grid the $u = (u_i, u_j, u_z)$ Fourier transformation is computed and denoted as $\widehat{u} = (u_{ki}, u_{kj}, u_{kz})$. The velocity spectrum tensor is computed as [112]:

$$E(\mathbf{k}) = \frac{1}{2} \left| \widehat{u}(\mathbf{k}) \cdot \widehat{u}^*(\mathbf{k}) \right|, \tag{4.35}$$

where $\widehat{U}^*$ is the complex conjugate of the transform velocity, and $\mathbf{k} = (k_i, k_j, k_z)$ is the wave number. Finally the energy spectrum $E(k)$ is obtained as

$$E(k) = 4\pi k^2 \left\langle E(\mathbf{k}) \right\rangle, \tag{4.36}$$

where $\langle ... \rangle$ is an average over the thin spheric shell of radius $k = |\mathbf{k}|$.

The temporal evolution of the energy spectrum for both models (DNS and Smagorinsky) with $Re = 2 \times 10^3$ is shown in Fig. 4.16, along with the kinetic energy evolution for both models in Fig. 4.17. It can be observed that the energy spectra in Fig. 4.16 are in good agreement. It can also be seen that more energy is dissipated by the Smagorinsky model for high wave numbers. Fig. 4.17 shows the subgrid model to dissipate energy faster than the DNS model due to the modelled small scales.

The Courant number (CFL) defined in Eq. (4.14) is an important indicator of the stability of the method, Fig. 4.18 shows the time evolution of the Courant number for both models at $Re = 2 \times 10^3$. It can be observed that the DNS produces high error and gradients, with big instability in the Courant number as it tries to resolve the many small scales in the turbulent flow, however the Courant number stabilised after time $t = 20$, on the the other hand Smagorinsky model did not suffer of such instability and the Courant number is decreasing in relatively stable manner.

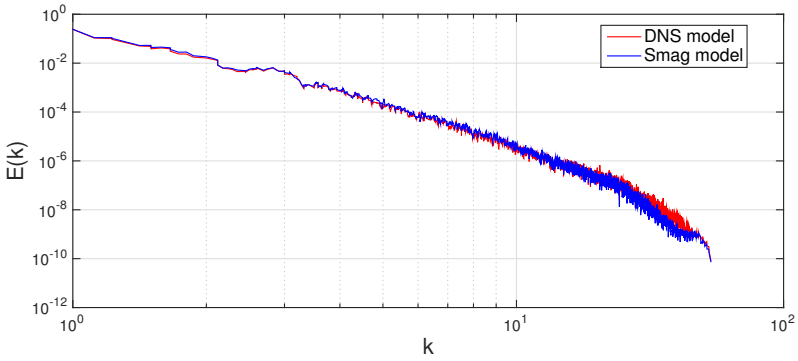


Figure 4.16: The energy spectra, for both the DNS and the Smagorinsky model, with $Re = 2 \times 10^3$ using the hrSPH method. it can be observed that the energy spectra are in good agreement. It can also be seen that more energy is dissipated by the Smagorinsky model for high wave numbers.

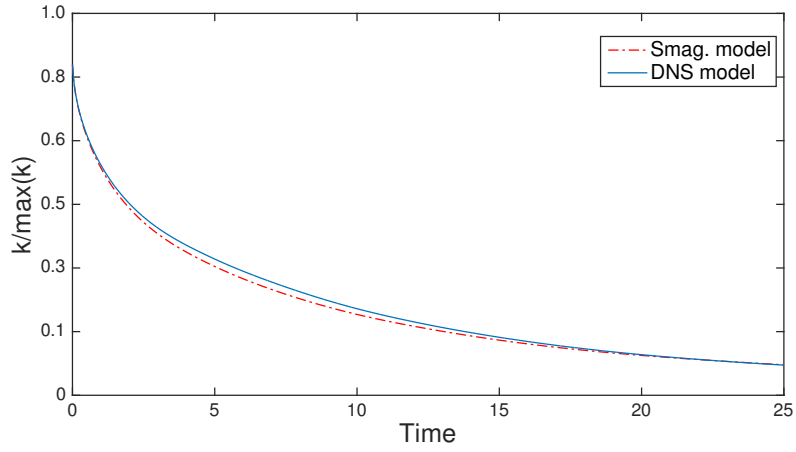


Figure 4.17: The kinetic energy evolution in time, evaluated for both the DNS and the Smagorinsky model, with $Re = 2 \times 10^3$ using the hrSPH method. Here the subgrid model dissipate energy faster than the DNS model due to the modelled small scales.

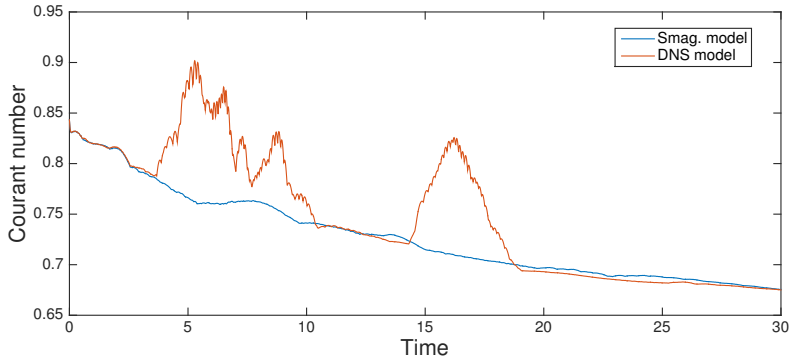
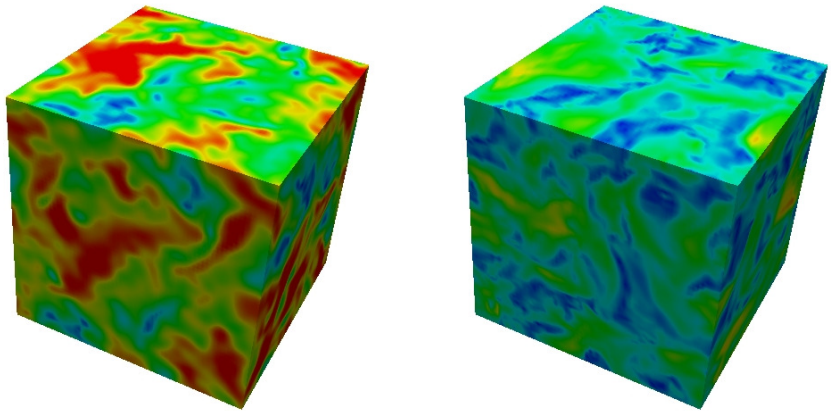


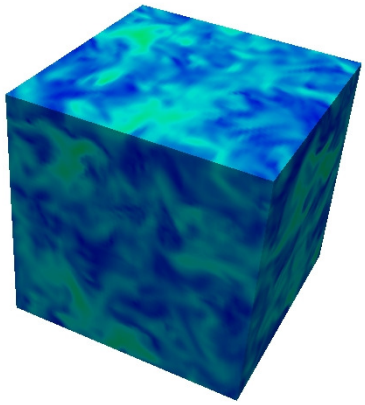
Figure 4.18: The Courant number evolution for the Smagorinsky and DNS model with $Re = 2 \times 10^3$. It can be observed that the DNS produces high error and gradients, with big instability in the Courant number as it tries to resolve the many small scales in the turbulent flow

Fig. 4.19 (a) shows the initialised velocity at time $t = 0$ and the simulated velocity magnitude decay Fig. 4.19 (b),(c), and (d) for Reynolds number $Re = 6 \times 10^3$. The hrSPH method with Smagorinsky model is used with remeshing every time step, as we find that for flow with strong vorticity the particle distribution becomes distorted and the particles tend to clustered so that remeshing in each time step is necessary.

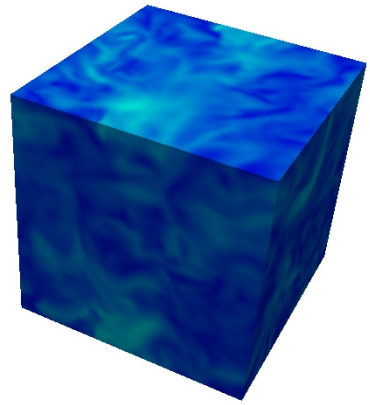


(a) time = 0

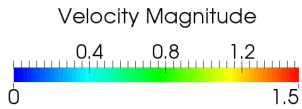
(b) time = 10



(c) time = 30



(d) time = 60



(e)

Figure 4.19: Velocity magnitude decay of the isotropic turbulence case

The energy spectrum calculated as in Eq. (4.35), and the dissipation of the energy agrees with the Kolmogorov $-5/3$ profile [70]. The temporal

evolution of the energy spectrum from the initial state to the stationary state at time= 60 is shown in Fig. 4.20.

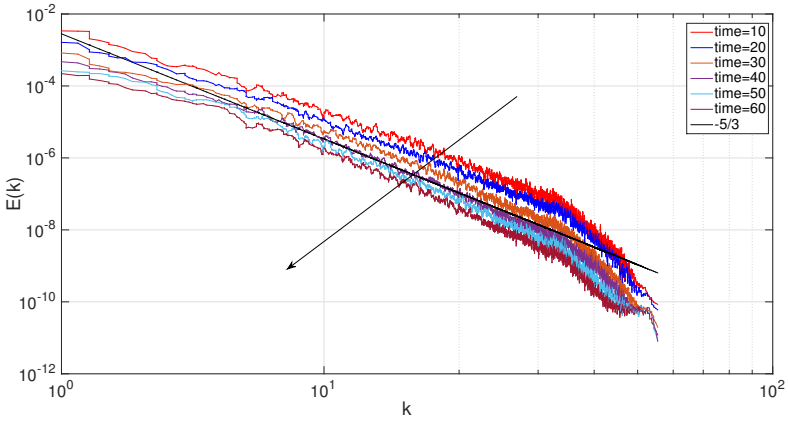


Figure 4.20: Time evolution of the energy spectra, evaluated at different time during the simulation, $Re = 6\times10^3$ using the hrSPH method with Smagorinsky mode, along with the Kolmogorov $-5/3$ profile.

4.5 Summary

We presented a hybrid remeshed smoothed particle hydrodynamics method (hrSPH), taking advantage of the Lagrangian advection, and the finite difference efficiency by computing the differential operators on the mesh.

Two models were used a DNS model, and a Smagorinsky model. We verified our method through several benchmarks, the hrSPH is able to resolve the flow with varying Reynolds number from 1 up to 10^4. The method showed a third order converging for the Taylor Green flow case.

As a result of the Lagrangian advection that the method enjoys, we were able to resolve the double thin shear layer without producing the spurious vortical structure with a coarser mesh than what other studies suggested. And finally, the hrSPH method resolved the three-dimensional isotropic turbulence flow with high Reynolds number on a coarse mesh using Smagorinsky model.

4.6 Acknowledgements

The author is grateful to the Danish Centre for Scientific Computing, for their computational resources. MAN Diesel & Turbo and The Danish

Council for Independent Research Technology and Production Sciences (FTP) for their finance support to the project. John Hopkins University (JHU) Turbulence Database Cluster, for their raw turbulence data that are used in our simulation. Professor Ivo Sbalzarini from The Max Planck Institute of Molecular Cell Biology and Genetics (MPI-CBG) for his helpful discussions and supervision.

Chapter 5

A Brinkman penalisation method for hybrid remeshed smoothed particle hydrodynamics method of compressible flow

5.1 Background

Insight in the response of many engineering applications can be obtained using numerical simulations of viscous flows around solid obstacle.can be distinguished to treat complex geometries: (i) the body-fitted grid (BF) method [102], and (ii) the immersed boundary (IB) method [94, 88].

The BF method proposes to generate grids associated with complex boundaries. Consequently, boundary conditions are easily specified. In order to achieve sufficiently accurate results for flows with high Reynolds numbers, a fine or grid is required for the boundary layer. However generating a good quality fine mesh can be cumbersome. . Fine meshes furthermore

entail substantial computational efforts. In moving boundaries cases the simulation setup becomes more complex and expensive as a result of the grid generation process, and the interpolation process of the solution to the new mesh at each computational time step, the BF method can therefore be complex to implement while its computational costs are significant.

Peskin [16] introduced the IB method as a new approach to study the flow around heart's valves. Ever since, it has developed in an alternative approach to the BF method to describe the interaction of fluids with complex geometries. This method is called immersed boundary (IB) method.

The IB method consists of a mathematical formulation and a numerical scheme [16]. The mathematical formulation is a mixture between Eulerian and Lagrangian variables, related by an interaction equation in which the Dirac delta function plays a notable role. In the numerical scheme, the Eulerian variables are defined on a fixed grid, whereas the Lagrangian ones are defined on a curvilinear grid that moves freely through the fixed grid. Refinement of the curvilinear grid is not required.

Complex boundaries are modelled as immersed elastic boundaries in IB methods. They interact with the surrounding fluids by solving the Navier-Stokes equations, whilst modifying the momentum equation. Peskin [16] showed that the boundaries do not need to be massless and that the fluid's density does not need to be uniform.

For rigid obstacle problems Lai [68] extended the method by having a stiff spring with restoring force represent the elastic media. This method was later extended by using a feedback force to represent the immersed boundary for solid obstacle problems [18, 67].

The BF and the IB methods have some disadvantages. Both use an explicit time-stepping scheme for instance. This entails that small time steps are required, compromising the efficiency. They are also restricted to non-adaptive grids, making them inefficient for flows with high Reynolds numbers. Lia [68] showed that, in order to mimic the real situation of flow around a solid obstacle, the computational domain must be relatively large compared to the solid obstacle. Moreover, no convergence proof exists for both methods.

Peskin's immersed boundary method uses external forces to simulate the boundaries of the computational domain. Cartesian grid methods [109, 49, 114] and ghost cell immersed boundary methods [30] are IB methods which in contrast to Peskin's method, enforce the boundary conditions directly upon on the immersed boundaries.

The Cartesian grid method was extended for compressible flow simulations by Ghias [87], simulating a compressible flow around a circular cylinder and airfoil at high Reynolds number, by modifying the discretised equation near the immersed boundaries. A drawback of this approach is not

taking in account the acoustic wave reflection and transmission between the solid and fluid at the interface area, which is important for shock wave simulations.

In this work we are propose another variant of the immersed boundary methods, in which the boundary conditions are imposed by penalising the governing equations by adding a penalised term. This is often referred to as the Brinkman penalisation.

The Brinkman penalisation was originally proposed by Arquis and Caltagirone [22], Its main idea is to model the complex obstacle as porous media with porosity ϕ and viscous permeabilityα approach to zero.

The Brinkman method has an important advantage: its error bound can be estimated in term of the penalisation parameter [76]. The boundary conditions are enforced to a specific precision with no need to modify the numerical method or adapt the grid. Angot [75] showed that for incompressible Navier-Stokes equations, the method converges to the exact solution as the penalisation parameter approaches zero.

We employ Liu's [86] extension of the Brinkman penalisation for compressible flows. We introduce an implicit boundary approach for the viscous flow with high a Reynolds number using the hybrid remeshed smoothed particle hydrodynamics method (hrSPH).

In the following sections we will present the porous media equations, followed by the Navier-Stokes equation for the hrSPH method coupled with the Brinkman penalisation. We will then verify the presented method for a several benchmark problems, presenting the high accuracy of the method. Then we close with conclusions and recommendations for future work.

5.2 Porous media equations

In this section we will introduce the porous medium equations for compressible flow in order to obtain the corresponding Navier-Stokes equation for the hrSPH method with Brinkman penalty parameters. The main idea of the Brinkman penalisation method [22] is to model the solid obstacle as a porous medium. The governing equations for compressible fluids and penalised Navier-Stokes for the porous medium are solved simultaneously, since no interface conditions are required. For more details about porous media, the reader referred to [7, 65, 61].

5.2.1 Porous media properties

A porous medium is a material containing pores. The skeletal portion of the material is often called the "matrix", and is interconnected by pores.

The pores are typically filled with a fluid (liquid or gas). A porous medium is characterised typically by its porosity ϕ, and permeability α. The flow through permeable media is characterised by two length scales: the size of the pores d, and the macroscopic length L. The porosity ϕ is the fraction of the volume of connected pores over the total volume, allowing the fluid to pass through. The permeability α is a measure of the ability of a material to transmit fluid and is proportional to ϕd^2.

5.2.2 Continuity equation

The continuity equation for porous media reads:

$$\frac{\partial \rho}{\partial t} = -\frac{1}{\phi}\nabla \cdot (\rho \mathbf{v}), \tag{5.1}$$

where ρ is the interstitial fluid density, $\frac{\partial}{\partial t}$ is the derivative to the time, and v is the Darcy velocity $\mathbf{v} = (v_1, v_2, v_3)$, $\mathbf{v} = \mathbf{u}\,\phi$. As $\phi \ll 1$, where $\|\mathbf{v}\| < \|\mathbf{u}\|$, $\mathbf{u} = (u_1, u_2, u_3)$ is the interstitial velocity of the fluid.

5.2.3 Darcy's law, Brinkman equation and extensions

The first Darcy's law reads:

$$\mathbf{v} = -\frac{\alpha}{\mu}\nabla p, \tag{5.2}$$

where μ is the dynamic viscosity, and p is the intrinsic pressure. In order to meet the no-slip boundary condition an additional viscous therm can be added as follows [15]

$$\nabla p = \frac{\mu}{\alpha}\,\mathbf{v} + \mu \nabla^2 \mathbf{v}, \tag{5.3}$$

where the first viscous term is the Darcy's term, and the second is the Laplacian term of the Navier-Stokes equation. Wooding [9] extended this expression in order to make it similar to the Navier-Stokes equation, as follows:

$$\rho\left[\phi^{-1}\frac{\partial \mathbf{v}}{\partial t} + (\phi^{-1}\mathbf{v}\nabla)(\phi^{-1}\mathbf{v})\right] = -\nabla p - \frac{\mu}{\alpha}\mathbf{v}, \tag{5.4}$$

which was in turn extended to Brinkman equation by Vafai and Tien [52]:

$$\rho\left[\phi^{-1}\frac{\partial \mathbf{v}}{\partial t} + (\phi^{-1}\mathbf{v}\nabla)(\phi^{-1}\mathbf{v})\right] = -\nabla p - \frac{\mu}{\alpha}\mathbf{v} + \mu\nabla^2\mathbf{v}. \tag{5.5}$$

Finally, Eq. (5.5) can be rewritten as a volume-averaging method as [52]:

$$\frac{1}{\phi}\frac{\partial \rho v_i}{\partial t} = -\frac{1}{\phi}\frac{\partial}{x_j}(\rho\phi^{-1}v_i v_j) - \frac{\partial p}{x_i} + \mu\frac{\partial^2 v_i}{\partial x_j^2} - \frac{\mu}{\alpha}v_i, \tag{5.6}$$

Beck [53] showed that in Eq. (5.5) the convection term $(\phi^{-1}\mathbf{v}\nabla)(\phi^{-1}\mathbf{v})$ is inconsistent with the no-slip boundary conditions, and it was subsequently dropped by Nield [7] later.

Since Eq. (5.5 entails that the momentum decays with the order of $\exp[-(\phi/\alpha)t]$, neglecting the coefficient ϕ^{-1} on the lefthand side hardly affects the solution. Consequently, the momentum remains to decay sufficiently fast [86].

Liu and Vasilyev [86] presented a simplified momentum equation with a Brinkman penalisation for compressible flows. Their simplification is possible since the penalisation term results in a significant damping of the momentum inside the porous media whilst the no-slip boundary conditions are satisfied.

5.2.4 Brinkman penalisation for compressible flow

Starting with the Navier-Stokes equations for compressible flow Eq. (5.7, 5.8),

$$\frac{D\rho}{Dt} = -\rho\frac{\partial u_i}{\partial x_j} \tag{5.7}$$

and the conservation of momentum,

$$\rho\frac{Du_i}{Dt} = -\frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j} + \frac{\partial \tau_{ij}^{sgs}}{\partial x_j} \tag{5.8}$$

in which

$$\tau_{ij} = \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\delta_{ij}\frac{\partial u_k}{\partial x_k}\right) \tag{5.9}$$

where $\frac{D\diamond}{Dt} = \frac{\partial\diamond}{\partial t} + (u\cdot\bigtriangledown)(\diamond)$ denotes the material derivative, u_i is the velocity, x is the position, p is the pressure, ρ is the density, τ_{ij} is the shear stress, μ is the dynamic viscosity, δ_{ij} is the Kronecker delta, and τ_{sgs} is the sub-grid stress tensor, which is zero for direct numerical simulation.

In the following, the viscous compressible flow around a set of obstacles O_i is numerically simulated, whilst the velocity on the surface of the obstacle must meet the no-slip boundary condition:

$$\mathbf{u} = \mathbf{U_{oi}} \quad \text{on } O_i, \forall i, \tag{5.10}$$

where $\mathbf{U_{oi}}$ is the velocity in the obstacle O_i.

To specify no-slip boundary conditions without directly imposing Eq. (5.10), we can follow Angot [76] by adding penalty terms to the momentum equation. This extension results in a loss in mass and energy of the waves reflectedfrom the obstacle, producing wrong simulation results. Liu and Vasilyev [86] combined both the Navier-Stokes and porous media equations, resulting in a Brinkman penalisation method for compressible flows:

$$\frac{D\rho}{Dt} = -\left[1+\left(\frac{1}{\phi}-1\right)\chi\right]\rho\frac{\partial u_i}{\partial x_j} \tag{5.11}$$

$$\rho\frac{Du_i}{Dt} = -\frac{\partial p}{\partial x_i}+\frac{\partial \tau_{ij}}{\partial x_j}+\frac{\partial \tau_{ij}^{sgs}}{\partial x_j}-\frac{\chi}{\eta}(u_i-U_{oi}) \tag{5.12}$$

The system is closed with the equation of state:

$$p=\rho\, c^2, \tag{5.13}$$

where ϕ is the porosity, $\eta=\alpha\phi$ is the normalised viscous permeability. Note that $0<\phi\ll 1$, and $0<\eta\ll 1$. χ is the solid mask, and defined as,

$$\chi_i(x)=\begin{cases}1 & \text{if } x\in O_i,\\ 0 & \text{Otherwise.}\end{cases} \tag{5.14}$$

In Eq. (5.14) the solid mask χ is defined that it is 1 inside the solid and 0 in the fluid.

To improve the numerical accuracy of the rate-of-change of the momentum, χ can be built via a polynomial "quasi-step" function, so that it varies smoothly from 0 to 1. The step function is sketched in Fig. 5.1.

The step function χ varies smoothly over the smooth interval of width L normal to the surface, coinciding to a fixed number of mesh cells. The step function is a function of the signed distance to the solid surface, with 1 at the solid and smoothly goes to 0 at the solid surface. The polynomial step function is continuous and differentiable, a second order over the first and last $L/4$ of the interval, and first order in the intermediate region.

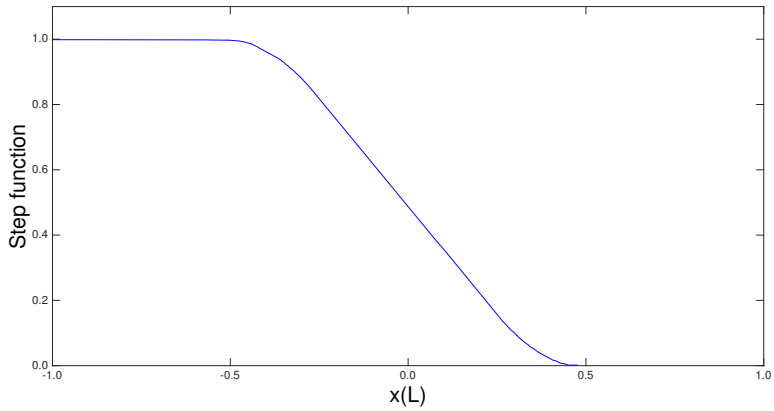


Figure 5.1: The polynomial step function as a function of the signed distance to the solid surface (second order over the first and last $L/4$, and first order in the intermediate region).

5.3 Brinkman penalisation with hrSPH method

After defining the computational domain, and initialising the mesh field on the mesh, the solid geometry is read as stereolithography triangulation (STL) file format containing a triangulated surface representation of an object through triangles described by three corner points and a normal, this file is called an STL patch. Then the χ fields on the nodes are initialised as a smooth step function by intersecting the computational mesh with the triangulated surface and initialising, the desired obstacle velocity U_0 is initialised. Subsequently, the hrSPH method is used directly without altering the numerical scheme; it intrinsically deals with the term $\frac{\chi}{\eta}(u - U_o)$ (the penalisation field).

5.4 Verification for Brinkman penalisation with hrSPH method

To validate the hrSPH method with Brinkman penalisation, we test a series of benchmark problems:

- Flow through a channel (Poiseuille flow); to verify the implicit boundary approach using the hrSPH method solution with the classical analytical Poiseuille flow solution.

- Lid-driven cavity; here we simulate the moving top wall with a constant velocity $U_{max} = 1$ as $U_o = 1$, and we simulate the viscous flow with high Reynolds number to validate the method.
- Compressible shock wave; a real compressible flow with high Mach number.

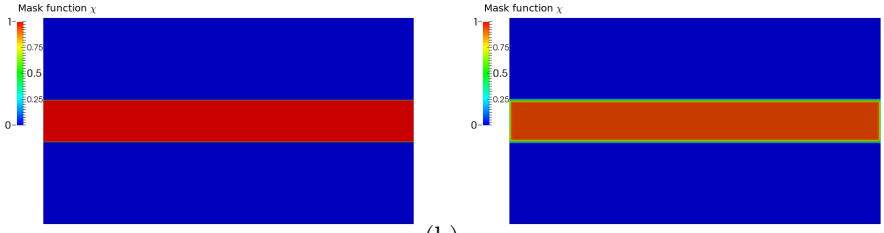
5.4.1 Poiseuille flow

The Poiseuille flow is a classical test case for laminar flow inside a pipe for which the analytical solution is known.

We study the flow throw a 2D channel. The computational domain is a periodic rectangle domain with length L and width $L/2$. At the centre of the domain a solid wall is represented as a penalised region, where the STL patch (triangulated surface) with length L and width $L/5$ is located as shown in Fig. 5.2.

Simulating the flow through a channel with a solid wall in the centre of the computational domain is the simplest way, and requires no additional work to impose symmetry. The upper part represents the first half of the channel, where the lower part is the other half. The simulation is started with a uniform density ρ and zero velocity $U = (u, v, w)$, external force in x-direction, with $\phi = 1 \times 10^{-2}$, and $\alpha = 2 \times 10^{-2}$.

Several computational domain resolution were tested to study the spatial convolution. We start with 32×32 particle and stop with 256×256 particles, resulting in mesh spacings of $\Delta x/L = 0.28, 0.14, 0.07, 0.035$. Note that the resolution of the triangulated surface (STL patch) can remain the same, even though the resolution of the computational mesh varies.



(a) (b)

Figure 5.2: The computational domain for the 2D Poiseuille flow with the penalised area as a Heaviside mask function (a) (χ), and (b) as a smooth step mask function (χ).

The simulated Poiseuille flow using the hrSPH method with Brinkman penalisation at different resolutions is shown in Fig. 5.3 for different res-

olutions of the computational grid, where u/U_{max} is extracted over the y-axis from the top of the channel till the centre of the penalised area. The Brinkman penalisation is able to reduce the velocity to zero. The results become more accurate for a decrease in the mesh resolution.

is shown in Fig. 5.3 for different resolutions of the computational grid, where u/U_{max} is extracted over the y-axis from the top of the channel till the centre of the penalised area. The Brinkman penalisation is able to reduce the velocity to zero. The results become more accurate for a decrease in the mesh resolution.

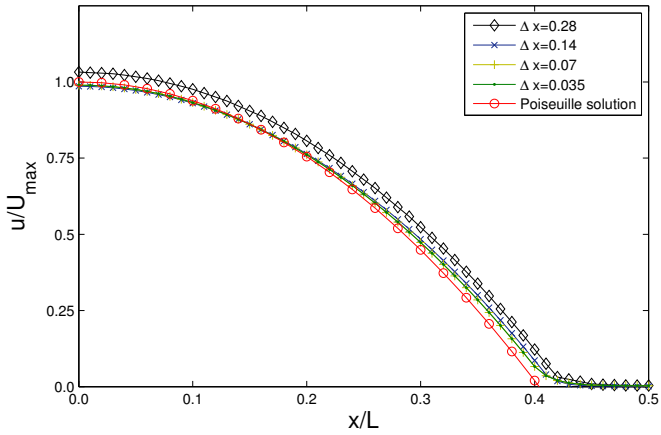


Figure 5.3: The normalised velocity magnitude of the Poiseuille flow with different resolution along with the analytical solution for the hrSPH method with Brinkman penalisation, the surface of the penalised area successfully managed to represent the required no-slip boundary condition at the channel surface at $x/L \approx 0.4$, forcing the normalised velocity u/U_{max} to drop to ≈ 0 as a result of the penalty term.

Fig. 5.3 illustrates how the surface of the penalised area successfully managed to represent the required no-slip boundary condition at the channel surface at $x/L \approx 0.4$, forcing the normalised velocity u/U_{max} to drop to ≈ 0 as a result of the penalty term. Note that only two cells are required to represent the smooth stepping in the solid surface. Different mesh resolutions are used to study the Poiseuille flow cf. Fig. 5.3, with finer resolution the solution is converging closer to the exact solution, till a level of refinement, and after that refining the mesh more does not affect the solution.

In Fig. 5.4 we show the effect ofof the smoothening of the penalised mask function χ as a Heaviside function, and as a signed distance smooth function, representing successfully the no-slip boundary condition.

The smooth function velocity profile is stepping smoothly toward the wall (channel wall) and needs around 2 cells inside the penalised area to represent the no-slip boundary condition. On the other hand the Heaviside function represents the no-slip condition approximately at the channel wall.

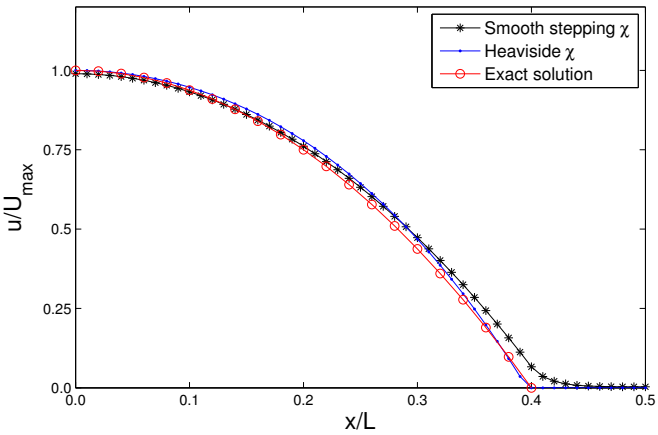


Figure 5.4: The normalised velocity magnitude of the Poiseuille flow simulated using a heaviside mask function (χ), and a smooth step mask function (χ) to represent the penalised term,, together with the analytical solution.

5.4.2 Flow around a cylinder

In the next test case we study the flow around a cylinder as presented by Morris [83]. The computational domain is a periodic square $L \times L$, and the cylinder is placed at the centre of the domain, with radius $R = L/5$ as in Fig. 5.5.

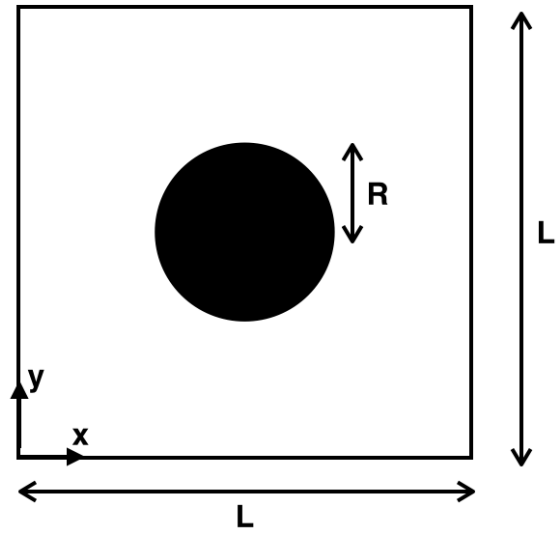


Figure 5.5: The periodic computational domain of the flow around a single solid cylinder in the centre.

The flow characteristic length Reynolds number $Re = 1$ and Mach number $Ma = 0.1$, with porosity $\phi = 1 \times 10^{-2}$, and permeability $\alpha = 2 \times 10^{-2}$. Two different resolutions were tested, with 100, and 200 grid point yield to $\Delta x = 0.1$, and $\Delta x = 0.05$ for the finer mesh in each direction. We are presenting the coarser mesh results in this section. The no-slip boundary conditions on the surface are imposed via Brinkman penalisation method, where the mask function χ is a smooth signed distance function.

Fig. 5.6 represents the velocity profile extracted over the y-axis at $x = L/2$, i.e. throughout the centre of the computational domain, normalised by the maximum velocity magnitude. Note how the mask function successfully manage to represent the no-slip condition around the cylinder surface, representing the cylinder as a porous medium. Fig. 5.7 represents the velocity magnitude.

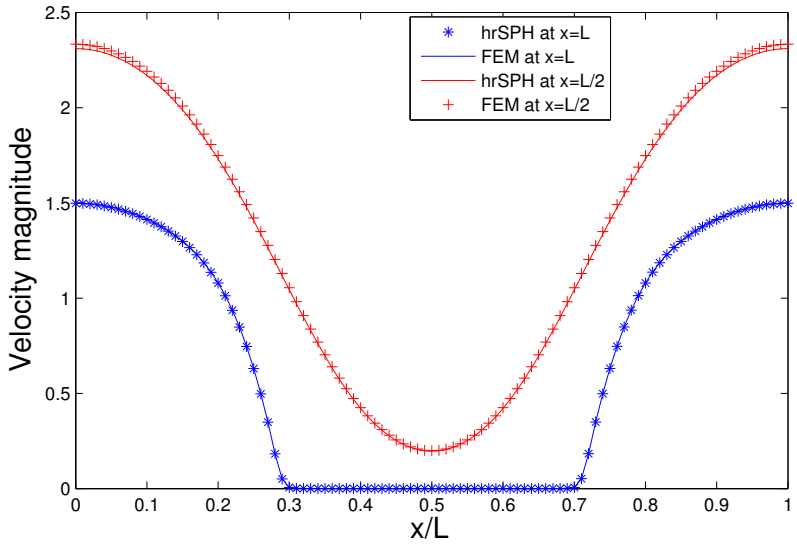


Figure 5.6: Axial velocity profile at $x = L/2$ (Path1), and $x = L$ (Path2).

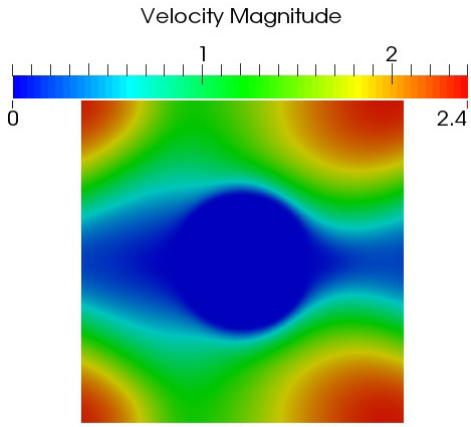


Figure 5.7: Velocity magnitude fields.

5.4.3 Lid-driven cavity

In the second benchmark problem we consider the lid-driven cavity problem using the hrSPH method with Brinkman penalisation. When simulating the lid-cavity two difficulties arise. First singularities at the top corners appear, due to the lid moving horizontally and the no-slip conditions on the vertical walls. The second difficulty is the high velocity gradients that arise at high Reynolds numbers.

The computational domain is a unit square with length L and the top wall is moving with a constant speed $U_{\max} = 1$. The computational domain is completely bounded by an STL patch to represent the domain walls using the Brinkman penalisation method Fig. 5.8. To simulate the moving top wall, an extra STL patch is introduced , which is given a constant speed $U_{\max} = 1$ as $U_s = 1$ which will represent the moving wall, as seen in Fig. 5.8.

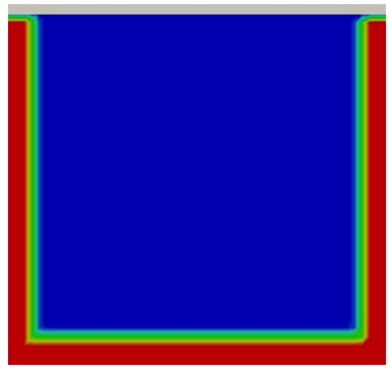


Figure 5.8: The Lid-driven cavity computational domain, with the bounded penalised walls represented as a step function. The moving top wall is presented in grey.

Three different Reynolds number $Re = (10^2, 10^3, \text{and } 10^4)$ are simulated with Mach number $Ma = 0.1$.

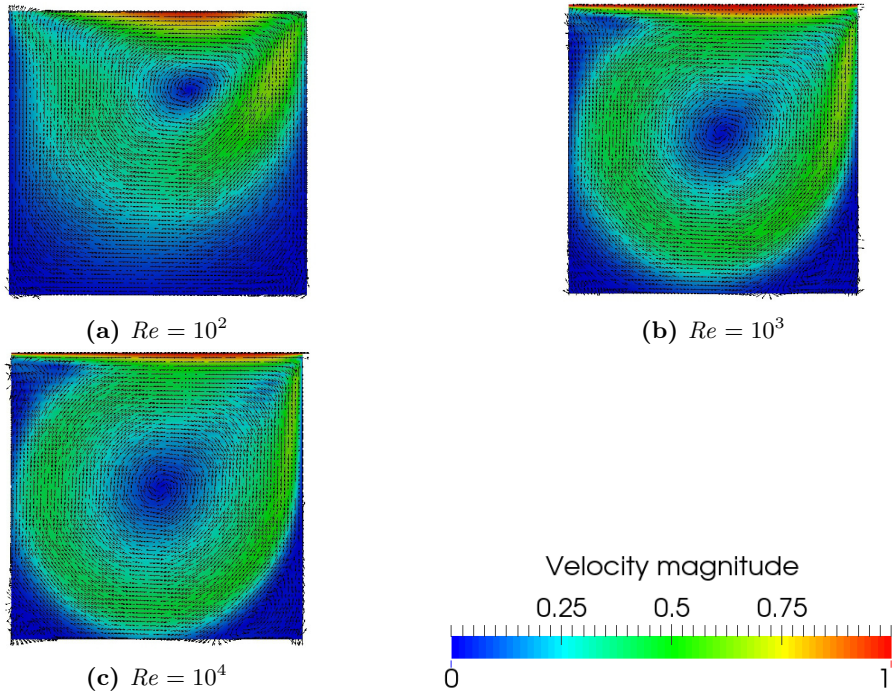
As no analytical solution for the Lid-driven cavity problem exists, we use

the results of Ghia [103] as a reference solution. Ghia used a highly-resolved multi-grid finite difference method with 257×257 grid. For the hrSPH method with Brinkman penalisation two grid resolutions are tested to capture the structure of the flow, 100×100 and 200×200 particles, yielding $\Delta x = 0.01, \Delta x = 0.005$. The results after a steady state occurs are extracted and compared with the reference solution. The steady state is reached when the total kinetic energy remains constant in time.

Fig. 5.9(a) presents the steady state field for $Re = 10^2$ with $\Delta x = 0.01$. The velocity magnitude ranges form zero (blue) to $U_{\max} = 1$ red, along with the velocity vector fields, which represent the structure of the flow. As a result of the shear force at the moving wall, a single vortex core is occurs in the top half of the domain. At the bottom the fluid is moves relatively slowly.

To compare the computed flows with those of Ghia we extract the velocity in the y-direction across a centreline in the x-direction ($U_v(x)$), and the velocity in the x-direction across a centreline in the y-direction ($U_x(y)$). In Fig. 5.10(a) the computed profiles for $Re = 10^2$ over the horizontal, and vertical centrelines. Both velocity profiles show a good agreement with Ghia's reference solutions [103].

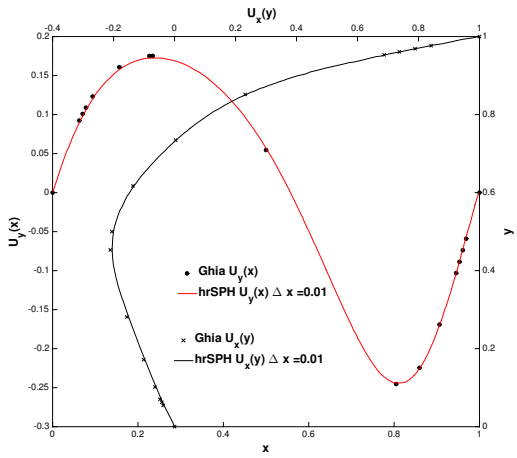
The velocity fields for the higher Reynolds number at $Re = 10^3$ and 10^4 are presented in Fig. 5.9(b), and Fig. 5.9(c). For an increase of the Reynolds number, the vortex core moves clearly to the centre of the domain, and the intensity of the vortex increases as well. The velocity also increases in the bottom part of the cavity. The coarse grid with ($\Delta x = 0.01$) failed to accurately predict the vortex location for both higher Reynolds number $Re = 10^3$ and 10^4, as can be seen in Fig. 5.10(b), and Fig. 5.10(c), where both velocity profiles ($U_x(y)$), ($U_v(x)$) are slightly shifted off the the reference solution. To capture the flow structure at $Re = 10^3$, a finer grid with $\Delta x = 0.005$ is used. Employing this grid, the solution shows a good agreement with Ghia's results as the velocity profiles illustrate.



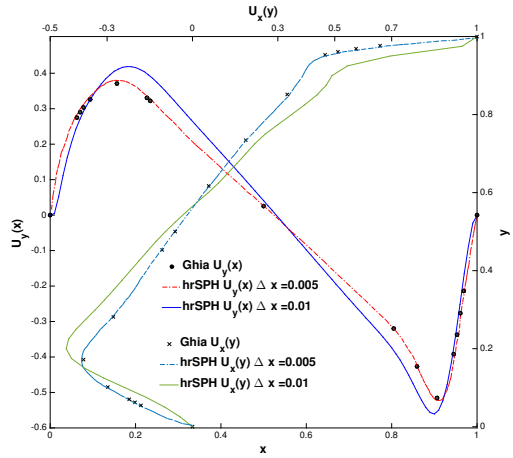
(a) $Re = 10^2$ **(b)** $Re = 10^3$

(c) $Re = 10^4$

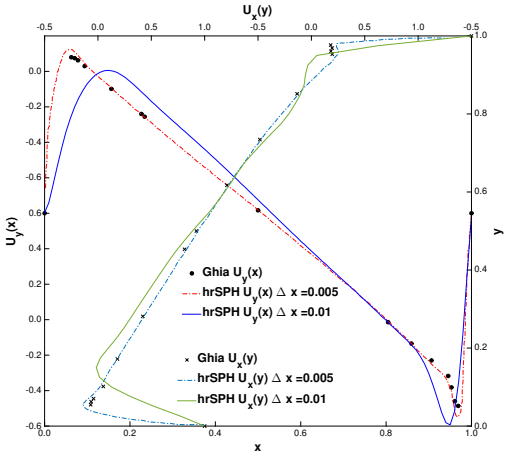
Figure 5.9: The velocity fields (magnitude and direction) computed for the lid-driven cavity problem for different Reynolds numbers.



(a) $Re = 10^2$



(b) $Re = 10^3$



(c) $Re = 10^4$

Figure 5.10: The velocity profiles for the lid-driven cavity problem compared to those of Ghia [103].

5.4.4 Compressible shock wave

To simulate a compressible flow exposed to a shock wave, a rectangular computational domain $[-L, 2L], [-L, L]$ with periodic boundary conditions, a penalised cylinder with radius $L/10$ at the centre of the computational domain is considered (see Fig. 5.12(a)). The initial conditions

are pressure perturbations p' resulting from a Gaussian distribution [86]:

$$p' = 10^{-3} \exp\left[-\ln(2)\left(\frac{(x-4)^2 + y^2}{0.04}\right)\right] \tag{5.15}$$

with the following initial density:

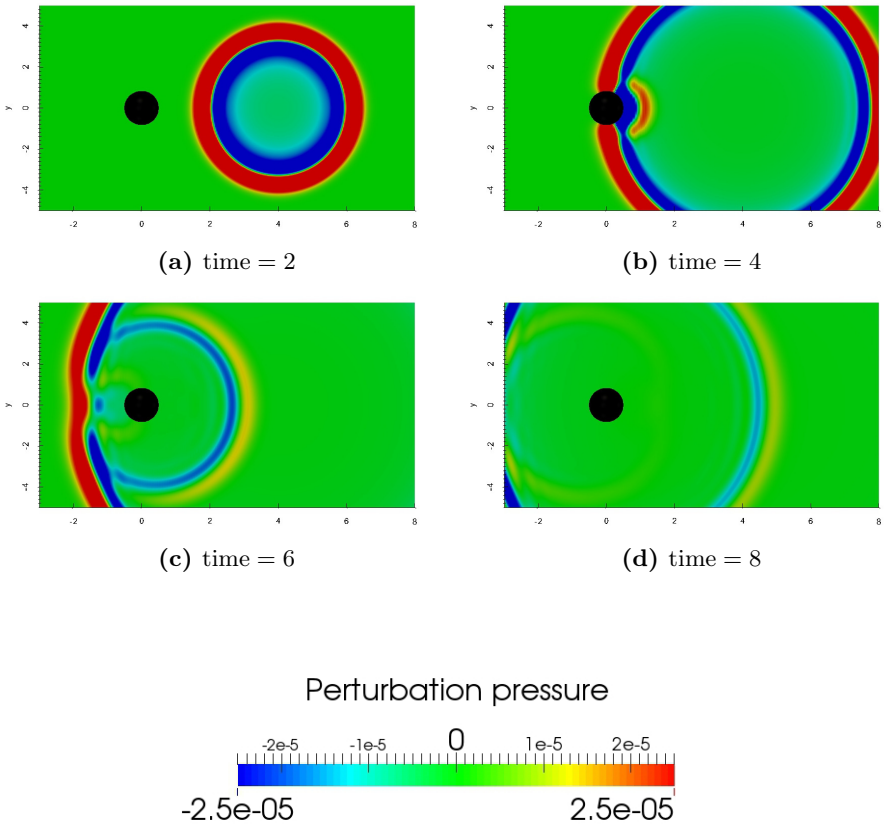
$$\rho = 1 + p'. \tag{5.16}$$

In this case $\phi = 0.02$, and $\alpha = 0.05$ are used.

The initial pressure perturbations form an acoustic wave, which propagates towards the solid cylinder at the origin of the computational domain, as illustrated in Fig. 5.11(a). When the wave front reaches the solid cylinder, it reflects , thereby producing a second wave propagating in the opposite direction. The main wave splits into two parts as a result of the interaction with the solid cylinder and continues to move in the same direction as shown in Fig. 5.11(b), which represents an instantaneous snapshot of the pressure perturbation at time $t = 4.0$. At time $t = 6.0$ the two parts of the main wave collide and merge on the left side of the solid cylinder producing a third wave, as shown in Fig. 5.11(c). In Fig. 5.11(d) the reflected wave and the main one both continue to propagate towards the boundaries and completely surround the solid cylinder.

The main issue in this example is the method's ability to capture the physical structure of the reflected wave and the third propagated wave. The quality of the reflected wave and the third propagated one strongly depends on how well the Brinkman penalisation method can map the solid cylinder in the computational domain and how well the no-slip condition is incorporated. Five points around the cylinder are marked as sample points (A-E in Fig. 5.12(a)), and the time history of the perturbation pressure is extracted. As the flow is symmetric around the x-axis, only the upper side of the cylinder is covered with the sample points, points A and E are important to observe the quality of the reflected wave and the third propagated wave.

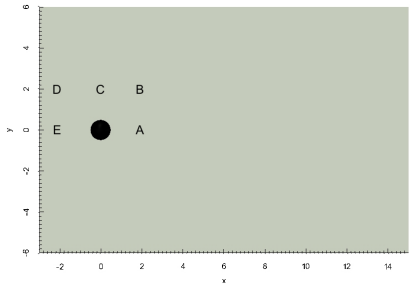
The numerical results are shown in Fig. 5.12(b-e), together with the exact solution (which is taken from [86]). Fig. 5.12(b) shows the perturbation pressure of the main wave at time $t \approx 2.0$ and, later the reflected wave at time $t = 4.0$ and later at time $t = 5.0$ for point B. Fig. 5.12(e) presents the perturbation pressure of the third wave after the two seperated waves merge and collide. All the numerical results show a good agreement with the exact solution, indicating the ability of the hrSPH method coupled with the Brinkman penalisation to capture the correct physics of the propagating waves, which is a results of the method's correct representation of the no-slip boundary condition.



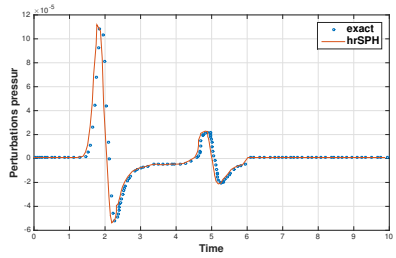
(a) time = 2 **(b)** time = 4

(c) time = 6 **(d)** time = 8

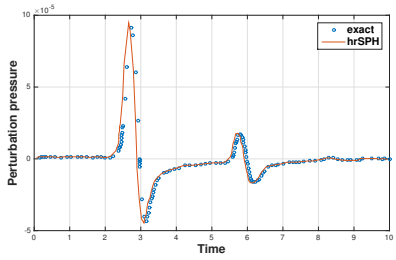
Figure 5.11: Pressure profiles computed for the compressible flow exposed to a shock wave at $t = 1.0$, $t = 4.0$, $t = 6.0$ and $t = 8.0$



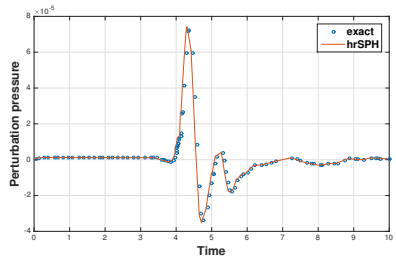
(a) The five sampling points in the domain.



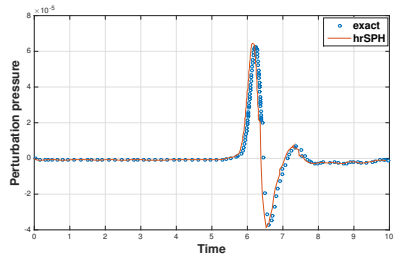
(b) Sampling point A.



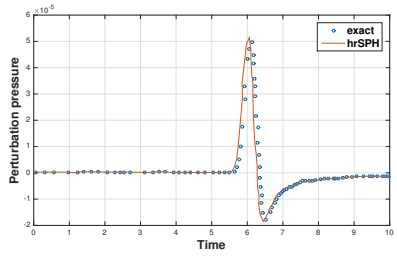
(c) Sampling point B.



(d) Sampling point C.



(e) Sampling point D.



(f) Sampling point E.

Figure 5.12: Compressible shock wave perturbation pressure responses for the compressible flow exposed to a shock wave at five points.

5.5 Summary

The coupled Brinkman penalisation technique with the hrSPH proved its efficient implementation, and good simulations results.

The technique is efficient to implement as there is no need to change the hrSPH numerical scheme, only a penalty term is added to the hrSPH Navier-Stokes equations, and a mask function is added on the mesh with

value = 1 wherever a solid obstacle is required in the computational domain.

Number of benchmark problem are presented to test the accuracy of the presented method. We start with a simple periodic laminar flow inside a pipe (Poiseuille flow),to flow around a cylinder. We tested more complex applications too, lid-driven cavity case is simulated with high Reynolds number, and finally we tested a case where a shock compressible wave is reflected from an obstacle and propagate throw out the computational domain. The results were found in good agreement with the exact solution and with the results that different methods provided.

Chapter 6

Conclusions

In the first part of this thesis we simulate the scavenging and swirling flow in the static geometry of a simplified model of a large diesel engine using LES with four different turbulence models. The results were compared with the experimental results.

It is found that the both the 'localised dynamic one-equation model' and the 'dynamic one-equation model' yield qualitatively and quantitatively good predictions of the mean axial and tangential velocity profiles. In contrast, the 'one-equation model' and the 'mixed-scale model' both fail to predict qualitatively the correct profile of the axial flow profiles.

We show that the position of the piston has a dramatic effect on simulated flow fields: displacing the piston such that the intake is covered by 50 % changes the time-averaged axial velocity profiles from a wake-like shape to a jet-like shape.

the LES simulation using the dynamic one equation eddy is able to predict the confined swirling flow with a reasonable accuracy. The piston position has a dominating influence on the in-cylinder flow, creating a qualitatively change in the flow topology.

In the second part we presented a remeshed particle-mesh method for the simulation of three-dimensional compressible turbulent flow. The method extends the classical smooth particle hydrodynamics method.

A uniform grid is introduced, where the quantities of the particles are interpolated onto the grid as a solution of the particle distortion. This hybrid particle-mesh method enjoys the advantage of the Lagrangian advection, and the finite difference efficient calculation of the pressure gradient, and turbulent diffusion. The method is capable of providing quality simulations while maintaining its robustness and adaptivity.

The method is based on the three dimensional pressure-velocity Navier-Stokes formulation, the algorithm is based on interpolating the impulse and mass of the particles into the grid, where the rate-of-change of fluid momentum is computed on the mesh, finally the interpolated rate of change in velocity is used to update the velocity and position of the particle.

The method has been validated through detailed simulations of the turbulent flow, two- and three-dimensional Taylor-Green flow, thin double shear layer, and three-dimensional isotropic turbulence. Two models were implemented direct numerical simulations, and Smagorinsky model. The results were found in good agreement with exact solutions, and the hrSPH exhibits a third order convergence in space for the Taylor Green flow case. As a result of the Lagrangian advection that the method enjoys, we were able to resolve the double thin shear layer without producing the spurious vortical structure with a coarser mesh than what other studies suggested. Finally the method resolved the the three-dimensional isotropic turbulence flow with high Reynolds number on a coarse mesh using Smagorinsky model.

The coupled Brinkman penalisation technique with the hrSPH proved its efficient implementation, and good simulations results.

The technique is efficient to implement as there is no need to change the hrSPH numerical scheme, only a penalty term is added to the hrSPH Navier-Stokes equations, and a mask function is added on the mesh with value = 1 wherever a solid obstacle is required in the computational domain.

Number of benchmark problem are presented to test the accuracy of the presented method. We start with a simple periodic laminar flow inside a pipe (Poiseuille flow),to flow around a cylinder. We tested more complex applications too, lid-driven cavity case is simulated with high Reynolds number, and finally we tested a case where a shock compressible wave is reflected from an obstacle and propagate throw out the computational domain. The results were found in good agreement with the exact solution and with the results that different methods provided. The results were found in good agreement with the exact solution and with the results that different methods provided.

Bibliography

[1] Bird G. A. *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*. Clarendon Press Oxford, Oxford, 1994.

[2] Dalrymple R. A. and Rogers B. D. Numerical modeling of water waves with the SPH method. *Coastal Engng.*, 53:141–147, 2006.

[3] Gingold R. A. and Monaghan J. J. Smoothed particle hydrodynamics: theory and application to non-spherical stars. *MONTH Notices Roy. Astron. Soc.*, 181:375–389, 1977.

[4] Leonard A. Vortex methods for flow simulation. *J. Comput. Phys.*, 37:289–335, 1980.

[5] Leonard A. Computing three-dimensional incompressible flows with vortex elements. *Annu. Rev. Fluid Mech.*, 17:523–559, 1985.

[6] Milton A. and Stegun I. *Handbook of Mathematical Functions With Formulas, Graphs and Mathematical Tables*. National Bureau of Standards. Applied Mathematics Series. 55, 1972.

[7] Nield D. A. and Bejan A. Convection in Porous Media. 1999.

[8] Obeidat A., Schnipper T., Ingvorsen K. M., Haider S., Meyer .K E., Stefan M., and Walther J. H. Large eddy simulations of the influence of piston position on the swirling flow in a model two-stroke diesel engine. *Int. J. Num. Meth. for Heat & Fluid Flow*, 24(2):325–341, 2014.

[9] Wooding R. A. Steady state free thermal convection of liquid in a saturated permeable medium. *J. Fluid Mech.*, 2:273–285, 1957.

[10] Yoshizawa A. and Horiuti K. A statistically-derived subgrid-scale kinetic energy model for the large-eddy simulation of turbulent flows. *J. Phys. Soc. Japan*, 54(8):2834–2839, 1985.

[11] Lucy L. B. A numerical approach to the testing of the fission hypothesis. *Astron. J.*, 82:1013–1024, 1977.

[12] Pope S. B. A more general effective-viscosity hypothesis. *J. Fluid Mech.*, 72(2):331–340, 1975.

[13] Pope S. B. Lagrangian pdf methods for turbulent flows. *Annu. Rev. Fluid Mech.*, 26:23–63, 1994.

[14] Lu Y. Y. Belytschko T. and Gu L. Element-Free Galerkin Methods. *Int. J. for Numer. Methods In Engng.*, 37:229–256, 1994.

[15] Brinkman H. C. A calculation of the viscous force exerted by a flowing fluid on a dense swarm of particles. *Appl. Sci. Res.*, 1(1):27–34, 1949.

[16] Peskin C. Flow patterns around heart valves: A numerical study. *J. Comput. Phys.*, 10:252–271, 1972.

[17] Drikakis D. and Smolarkiewicz P. K. On spurious vortical structures. *J. Comput. Phys.*, 172:309–325, 2001.

[18] Goldstein D., Handler R., and Sirovich L. Modeling a no-slip flow boundary with an external force field. *J. Comput. Phys.*, 105:354–366, 1993.

[19] Rossinelli D. and Koumoutsakos P. Vortex methods for incompressible flow simulations on the GPU. *Visual Comput.*, 24:699–708, 2008.

[20] Violeau D. and Issa R. Numerical modelling of complex turbulent free-surface flows with the SPH method: an overview. *Int. J. Numer. Meth. Fluids*, 53(2):277–304, 2007.

[21] de Castro Gouveia M., dos Reis Parise J. A., and Nieckele A. O. Numerical simulation of the fluid flow and heat transfer processes during scavenging in two-stroke engine under steady-state conditions. *J. Heat Transfer*, 114:383–393, 1992.

[22] Arquis E. and Caltagirone J-P. On the hydrodynamical boundary-conditions along a fluid layer porous-medium interface - application to the case of free-convection. *C. R. Acad. Sci. II*, 229(1):1–4, 1984.

[23] Hieber S. E. and Koumoutsakos P. A lagrangian particle level set method. *J. Comput. Phys.*, 210:342–367, 2005.

[24] Saffman P. G. *Vortex Dynamics.* Cambridge University Press, 1992.

[25] Cottet G-H. and Koumoutsakos P. *Vortex Methods – Theory and Practice.* Cambridge University Press, New York, 2000.

[26] Harlow F. H. and Welch J. E. Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with Free Surface. *Phys. Fluids*, 8(12):2182–2189, 1965.

[27] Schlichting H. *Boundary-Layer Theory.* McGraw Hill, Inc., 7. edition, 1979.

[28] Schweitzer P. H. *Scavenging of two-stroke cycle Diesel engines.* Macmillan Publishing Company, 1949.

[29] Takeda H., Miyama S. M., and Sekiya M. Numerical simulation of viscous flow by smoothed particle hydrodynamics. *Prog. Theor. Phys.*, 92(5):939–960, 1994.

[30] Tseng Y. H. and Ferziger J. H. A ghost cell immersed boundary method for flow in complex geometry. *J. Comput. Phys.*, 192:593–623, 2003.

[31] Babuska I. and Melenk J. M. The Partition of Unity Method. *Int. J. for Numer. Methods In Engng.*, 40(4):727–758, 1998.

[32] Issa R. I. Solution of the implicitly discretised fluid flow equations by operator-splitting. *J. Comput. Phys.*, 62:40–65, 1985.

[33] Taylor G. I. and Green A. E. Mechanism of the Production of Small Eddies from large ones. *Proc. R. Soc. Lond.*, 158(893):499–521, 1937.

[34] Bardina J., Ferziger J. H., and Reynolds W. C. Improved subgrid scale model for large eddy simulation. *AIAA Paper*, 80-1357, 1980.

[35] Chorin A. J. Vortex sheet approximation of boundary layers. *J. Comput. Phys.*, 27:428–442, 1978.

[36] Chorin A. J. and Bernard P. S. Discretization of a vortex sheet, with an example of roll-up. *J. Comput. Phys.*, 13:423–429, 1973.

[37] Cummins S. J. and Rudman M. An SPH projection method. *J. Comput. Phys.*, 152(2):584–607, 1999.

[38] Mansour J. *SPH and α-SPH: Applications and Analysis.* PhD. thesis, Monash University, October 2007.

[39] Monaghan J. J. Particle methods for hydrodynamics. *Comput. Phys. Rep.*, 3:71–123, 1985.

[40] Monaghan J. J. On the problem of penetration in particle methods. *J. Comput. Phys.*, 82(1):1–15, 1989.

[41] Monaghan J. J. Smoothed particle hydrodynamics. *Annu. Rev. Astron. Astrophys.*, 30:543–574, 1992.

[42] Monaghan J. J. Simulating free surface flows with SPH. *J. Comput. Phys.*, 110:399–406, 1994.

[43] Monaghan J. J. SPH compressible turbulence. *Mon. Not. R. Astron. Soc.*, 335:843–852, 2002.

[44] Smagorinsky J. General circulation experiments with the primitive equations, part I: the bacis experiment. *Monthly Weather Rev.*, 91(3):99–164, 1963.

[45] Stam J and Fiume E. Depicting fire and other gaseous phenomena using diffusion processes. *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*, pages 129–136, 1995.

[46] Watkins S. J., Bhattal A. S., Francis N., Turner J. A., and Whitworth A. P. A new prescription for viscosity in smoothed particle hydrodynamics. *A & A S*, 119:177–187, 1996.

[47] Binder K. and Heermann D. W. *Monte Carlo Simulation in Statistical Physics: An Introduction.* Springer, 1998.

[48] Chaniotis A. K., Poulikakos D., and Koumoutsakos P. Remeshed smoothed particle hydrodynamics for the simulation of viscous and heat conducting flows. *J. Comput. Phys.*, 182(1):67–90, 2002.

[49] Clarke D. K., Salas M. D., and Hassen H. A. Euler calculations for multielement airfoils using cartesian grids. *AIAA J.*, 24(3):353–358, 1986.

[50] George W. K., Beuther P. D., and Lumley J. L. Processing of random signals. In *Proceedings of the Dynamic Flow Conference*, pages 757–800, 1978.

[51] Liu W. K., Jun S., and Zhang S. Reproducing kernel particle methods. *Int. J. for Numer. Methods In Engng.*, 20(8–9):1081–1106, 1995.

[52] Vafai K. and Tien C. L. Boundary and inertia effects on flow and heat transfer in porous media. *Int. J. Heat Mass Transfer*, 24:195–203, 1981.

[53] Beck J. L. Convection in a box of porous material saturated with fluid. *Phys. Fluids*, 15:1377—-1383, 1972.

[54] Davidson L. Large-Eddy Simulations: A dynamic one-equation subgrid model for three-dimensional recirculating flows. In *11th International Symposium on Turbulent Shear Flow*, pages 3:26.1–26.6, 1997.

[55] Minion M. L. and Brown D. L. Performance of under-resolved two-dimensional incompressible flow simulations, II. *J. Comput. Phys.*, 138:734–765, 1997.

[56] Ta Phuoc L. Modèles de sous maille appliqúes aux ecoulements instationnaires décollés. In *Proceedings of the DRET Conference: Aérodynamique Instationnaire Turbulente-Aspects Numériques et Expérimentaux*, pages 1–38, Orsay, France, 1994. Lab. d'Informatique pour la Mecanique et les Sciences de l'Ingénieur. Centre Natonal de la Recherche Scientifique.

[57] Duarte C. A. M. Liszka T. J. and Tworzydlo W. W. hp-meshless cloud method. *Comp. Meth. Appl. Mech. & Engng.*, 139:263–288, 1996.

[58] Ellero M., Espa nol P., and Adams N. A. Implicit atomistic viscosities in smoothed particle hydrodynamics. *Phys. Rev. E*, 82:046702, 2010.

[59] Germano M., Piomelli U., Moin P., and Cabot W. H. A dynamic subgrid-scale eddy viscosity model. *Phys. Fluids*, 3:1760–1765, 1991.

[60] Haile J. M. *Molecular Dynamics Simulations. Elementary Methods.* John Wiley & Sons, 1992.

[61] Kaviany M. Principles of Heat Transfer in Porous Media. *Springer-Verlag*, pages –, 1991.

[62] Lesieur M., Métais O., and Comte P. *Large Eddy Simulations of Turbulence.* Cambridge University Press, 2005.

[63] Melenk J. M. and Babuska I. The partition of unity finite element method: Basic theory and applications. *Comp. Meth. Appl. Mech. & Engng.*, 139:289–314, 1996.

[64] Müller M., Charypar D., and Gross M. Particle-Based Fluid Simulation for Interactive Applications. *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animatio*, pages 154–159, 2003.

[65] Phillips O. M. Flow and Reactions in Permeable Rocks. *Cambridge University Press*, pages –, 1991.

[66] Robinson M. and Monaghan J. J. Direct numerical simulation of decaying two-dimensional turbulence in a no-slip square box using smoothed particle hydrodynamics. *Int. J. Numer. Meth. Fluids*, 70(1):37–55, 2012.

[67] Saiki E. M. and Biringen S. Numerical simulation of a cylinder in uniform flow: Application of a virtual boundary method. *J. Comput. Phys.*, 123:450–465, 1996.

[68] Lai M-C. and Peskin C. S. An immersed boundary method for formal second-order accuracy and reduced numerical viscosity. *J. Comput. Phys.*, 160:705–719, 2000.

[69] Atluri S. N. and Zhu T. A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics. *Comput. Mech.*, 22:117–127, 1998.

[70] Kolmogorov A. N. The Local Structure of Turbulence in Incompressible Viscous Fluid for Very Large Reynolds Numbers. *Dokl. Akad. Nauk SSSR*, 30:301–305, 1941.

[71] Syred N. A review of oscillation mechanisms and the role of precessing vortex core (PVC) in swirl combustion systems. *Prog. Energy Combust. Sci.*, 32:93–161, 2006.

[72] Touzot G. Nayroles B. and Villon P. The diffuse elements method. *C. R. Acad. Sci. II*, 313:133–138, 1991.

[73] Lucca-Negro O. and O'Doherty T. Vortex breakdown: a review. *Prog. Energy Combust. Sci.*, 27:431–481, 2001.

[74] Allen M. P. and Tildesley D. J. *Computer Simulation of Liquids*. Clarendon Press Oxford, Oxford, 1987.

[75] Angot P. Analysis of singular perturbations on the Brinkman problem for fictitious domain models of viscous flows. *Math. Meth. Appl. Sci.*, 22:1395–1412, 1999.

[76] Angot P., Bruneau C.-H., and Fabrie P. A penalization method to take into account obstacles in incompressible viscous flows. *Numer. Math.*, 81:497–520, 1999.

[77] Chatelain P., Cottet G.-H., and Koumoutsakos P. PMH: Particle-mesh hydrodynamics. *Int. J. Mod. Phy. C*, 18(4):610–618, 2007.

[78] Koumoutsakos P. *Direct Numerical Simulations of Unsteady Separated Flows Using Vortex Methods*. Ph.d. thesis, California Institute of Technology, 1993.

[79] Koumoutsakos P. Inviscid axisymmetrization of an elliptical vortex ring. *J. Comput. Phys.*, 138:821–857, 1997.

[80] Koumoutsakos P. Multiscale flow simulations using particles. *Annu. Rev. Fluid Mech.*, 37:457–487, 2005.

[81] Koumoutsakos P. and Leonard A. High-resolution simulation of the flow around an impulsively started cylinder using vortex methods. *J. Fluid Mech.*, 296:1–38, 1995.

[82] Morris J. P. Simulating surface tension with smoothed particle hydrodynamics. *Int. J. Numer. Meth. Fluids*, 33(3):333–353, 2000.

[83] Morris J. P., Fox P. J., and Zhu Y. Modeling low Reynolds number incompressible flows using SPH. *J. Comput. Phys.*, 136:214–226, 1997.

[84] Sagaut P. *Large Eddy Simulation for Incompressible Flows.* Springer, 1998.

[85] Wang P. *Large Eddy Simulation of Turbulent Swirling Flows and Turbulent Premixed Combustion.* PhD thesis, Lund Institute of Technology, Box 118 SE–221 00 Lund Sweden, 2005.

[86] Liu Q. and Vasilyev O. V. A Brinkman penalization method for compressible flows in complex geometries. *J. Comput. Phys.*, 227:946–966, 2007.

[87] Ghias R., Mittal R., and Lund T. S. A non-body conformal grid method for simulation of compressible flows with complex immersed boundaries. In *AIAA 42nd Aerospace Sciences Meeting and Exhibit*, pages –, January 2004.

[88] Mittal R. and Iaccarino G. Immersed boundary methods. *Annu. Rev. Fluid Mech.*, 37:239–261, 2005.

[89] Schiestel R. *Modeling and Simulation of Turbulent Flows.* John Wiley & Sons, 2007.

[90] Adami S., Hu X. Y., and Adams N. A. A transport-velocity formulation for smoothed particle hydrodynamics. *J. Comput. Phys.*, 241:292–307, 2013.

[91] Ghosal S., Lund T. S. Moin P., and Akselvoll K. A dynamic localization model for large-eddy simulation of turbulent flows. *J. Fluid Mech.*, 286:229–255, 1995.

[92] Haider S., Schnipper T., Obeidat A., Meyer K. E., Okulov V. L., Mayer S., and Walther J. H. PIV study of the effect of piston position on the in-cylinder swirling flow during the scavenging process in large two-stroke marine diesel engines. *J. Mar. Sci. Tech.*, 2012.

[93] Leibovich S. The structure of vortex breakdown. *Annu. Rev. Fluid Mech.*, 10:221–246, 1978.

[94] Peskin C. S. The immersed boundary method. *Acta Numerica*, pages 479–517, 2002.

[95] Rogallo R. S. and Moin P. Numerical simulation of turbulent flows. *Annu. Rev. Fluid Mech.*, 16:99–137, 1984.

[96] Shao S. and Lo E. Y. M. Incompressible SPH method for simulating Newtonian and non-Newtonian flows with a free surface. *Adv. Water Res.*, 26:787–800, 2003.

[97] Succi S. The Lattice Boltzmann Equation - For Fluid Dynamics and Beyond. *Clarendon Press*, pages –, 2001.

[98] I. F. Sbalzarini, J. H. Walther, B. Polasek, P. Chatelain, M. Bergdorf, S. E. Hieber, E. M. Kotsalis, and P. Koumoutsakos. A software framework for portable parallelization of particle-mesh simulations. *Lect. Notes Comput. Sc.*, 4128:730–739, 2006.

[99] I. A. Shirokov and T. G. Elizarova. Simulation of laminar-turbulent transition in compressible taylor–green flow basing on quasi-gas dynamic equations. *Journal of Turbulence*, 15(10):707–730, 2014.

[100] Craft T., Iacovides H., Launder B., and Zacharos A. Some swirling-flow challenges for turbulent CFD. *Flow, Turbul. Combust.*, 80:419–434, 2008.

[101] Liszka T. and Orkisz J. The finite-difference method at arbitrary irregular grids and its application in applied mechanics. *Computers & Structures*, 11:83–95.

[102] Warsi Z. U. A. Thompson J. F. and Mastin C. W. Boundary fitted coordinate systems for numerical solution of partial differential equations – a review,. *J. Comput. Phys.*, 47:1–108, 1982.

[103] Ghia U., Ghia K. N., and Shin C. T. High-Re solution for incompressible flow using the Navier-Stokes equations and a multigrid method. *J. Comput. Phys.*, 48:387–411, 1982.

[104] Schumann U. Subgrid scale model for finite difference simulations of turbulent flows in plane channels and annuli. *J. Comput. Phys.*, 18:376–404, 1975.

[105] Alekseenko S. V., Koubin P. A., Okulov V. L., and Shtork S. I. Helical vortices in swirl flow. *J. Fluid Mech.*, 382:195–243, 1999.

[106] Rees W. M. van, Leonard A., Pullin D. I., and Koumoutsakos P. A comparison of vortex and pseudo-spectral methods for the simulation of periodic vortical flows at high reynolds numbers. *J. Comput. Phys.*, 230:2794—-2805, 2011.

[107] Cleary P. W. and Monaghan J. J. Conduction modelling using smoothed particle hydrodynamics. *J. Comput. Phys.*, 148:227–264, 1999.

[108] Hockney R. W. and Eastwood J. W. *Computer Simulation Using Particles*. Institute of Physics Publishing, Bristol, PA, USA, 2. edition, 1988.

[109] Purvis J .W. and Burkhalter J. E. Prediction of critical mach number for store configurations. *AIAA J.*, 11:1170–1177, 1979.

[110] David C. Wilcox. *Turbulence Modeling for CFD*. DCW Industries, Inc., 1994.

[111] Li Y., Perlman E., Wan M., Yang Y., Meneveau C., Burns R., Chen S., Szalay A., and Eyink G. A public turbulence database cluster and applications to study Lagrangian evolution of velocity increments in turbulence. *J. of Turbomachinery*, 9(31):1–29, 2008.

[112] Shi Y., Ellero M., and Adams N. A. Analysis of intermittency in under-resolved smoothed-particle-hydrodynamics direct numerical simulations of forced compressible turbulence. *Phys. Rev. E*, 85:036708, 2012.

[113] Shen W. Z., Zhu W., and Sørensen J. N. Aeroacoustic computations for turbulent airfoil flows. *AIAA J.*, 47(6):1518–1527, 2009.

[114] Powell K. G. Zeeuw D. D. An adaptively refined cartesian mesh solver for the Euler equations. *J. Comput. Phys.*, 104:56–68, 1993.